HEWLETT PACKARD

BASIC/3000 Compiler

Reference Manual

BASIC/3000 Compiler

Reference Manual

HEWLETT PACKARD

HEWLETT-PACKARD COMPANY
5303 STEVENS CREEK BLVD., SANTA CLARA, CALIFORNIA 95050

PART NO. 32103-90001
PRODUCT NO. 32103A

Printed in U.S.A. 11/74

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another program language without the prior written consent of Hewlett-Packard Company.

Copyright © 1976 by HEWLETT-PACKARD COMPANY

LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the most recent date on which the technical material on any given page was altered. If a page is simply re-arranged due to a technical change on a previous page, it is not listed as a changed page. Within the manual, changes are marked with a vertical bar in the margin.

Pages	**Effective Date**
Title	Jun 1976
ii to v	Jun 1976
vii	Nov 1974
1-0 to 1-2	Nov 1974
2-1 to 2-5	Nov 1974
3-1 to 3-6	Nov 1974
4-1 to 4-6	Nov 1974

PRINTING HISTORY

New editions incorporate all update material since the previous edition. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The date on the title page and back cover changes only when a new edition is published. If minor corrections and updates are incorporated, the manual is reprinted but neither the date on the title page and back cover nor the edition change.

First Edition. . . . . Nov 1974
Update package #1 . . . . Jun 1976
Update #1 Incorporated . . . . Sep 1977

PREFACE

This manual explains how to use the BASIC/3000 Compiler. It is assumed that you are already well familiar with the BASIC/3000 programming language, as it is a requirement that your BASIC programs are debugged before attempting to compile. It is also assumed that you know how to prepare and run programs using the MPE system. Typical examples of use with MPE, however, are provided in this manual.

The following manuals are available for reference:

MPE Intrinsics Reference Manual (30000-90010)
MPE Commands Reference Manual (30000-90011)

For 3000 systems which are not Series II, the following difference in manuals should be noted:

- The *MPE/3000 Operating System Reference Manual* (32000-90001) is used in place of the *MPE Commands Reference Manual* or the *MPE Intrinsics Reference Manual.*

CONTENTS

Section I Page
INTRODUCING THE BASIC/3000 COMPILER
Purpose 1-1
Overview of Operation 1-1
Language Considerations 1-1
  Initializing Variables 1-1
  Non-BASIC Calls 1-1
  Number of Variables 1-1
  Program Size 1-2
  Program Names 1-2
  CHAIN and INVOKE 1-2
  COMMON Blocks 1-2
  FOR Statement 1-2
  Error Checking 1-2
  Nested BASIC 1-2

Section II Page
COMPILER COMMANDS
System Commands 2-1
  Compile Only 2-1
  Compile and Prepare 2-1
  Compile, Prepare, and Execute 2-1
  Prepare and Execute Only 2-1
  MAXDATA Parameter 2-2
Subsystem Commands 2-3
  Prompts, Comments, Continuation 2-3
  CONTROL Command 2-3
  ENTRY Command 2-4
  TITLE Command 2-5
  COMPILE Command 2-5
  EXIT Command 2-5

Section III Page
PROGRAMMING EXAMPLES
Single-Program Example 3-1
Segmented Programs Example 3-1

Section IV Page
INTERPRETING MESSAGES
Compile-Time Error Messages 4-1
Run-Time Warning Messages 4-3
Run-Time Error Messages 4-3

ILLUSTRATIONS

Title Page

Overview of Operations 1-0
Branching into FOR Loop 1-2
Compilation of Single Program 3-2
Compilation of Segmented Programs 3-4
Run-Time Environment of Segmented Example 3-6
Stack Traceback Example 4-4

TABLES

Title Page

Attributes of Compiler Files 2-2
Attributes of Run-Time Input and List Files 2-3

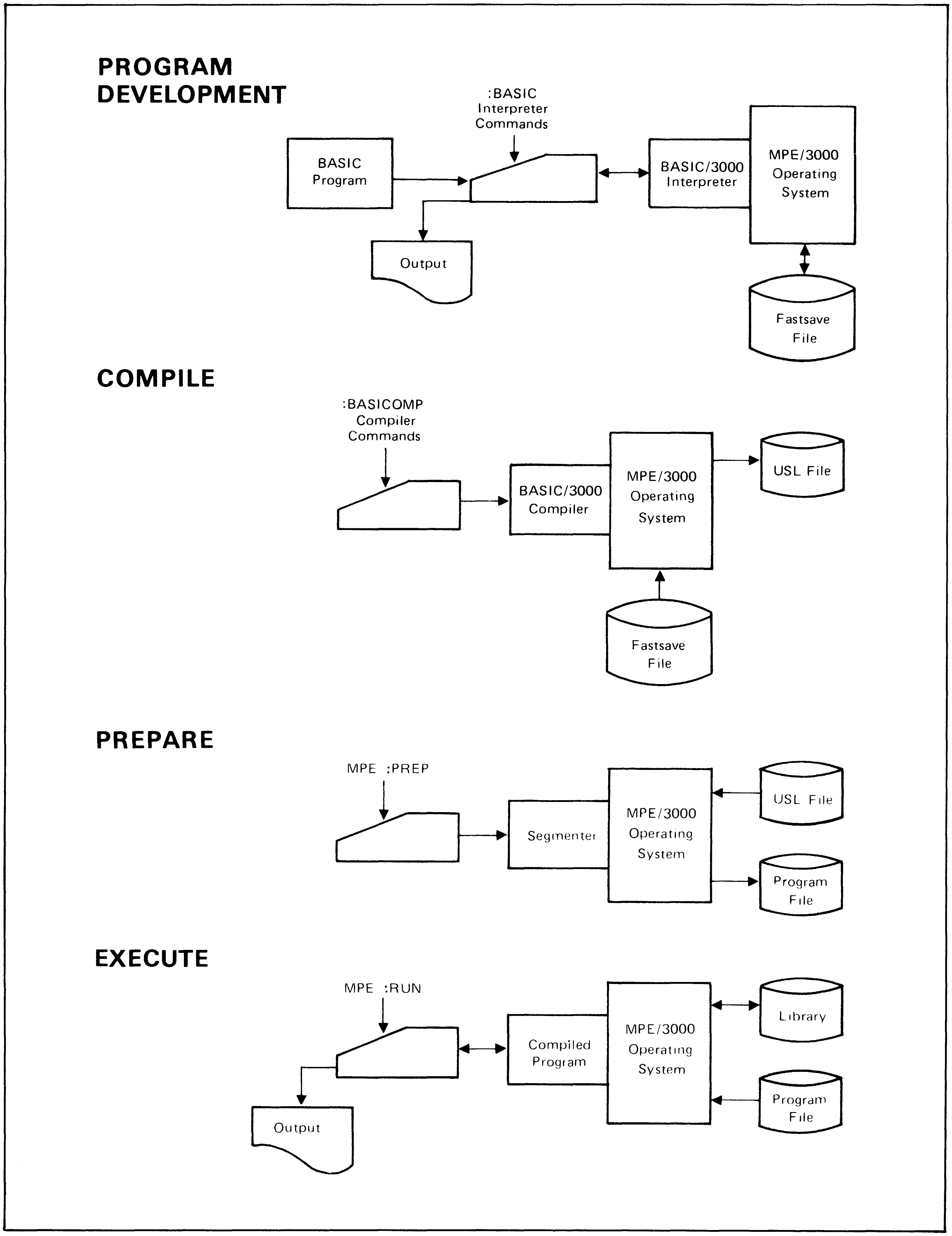


Figure 1-1. Overview of Operations

INTRODUCING THE BASIC/3000 COMPILER

SECTION I

1-1. PURPOSE

The HP 32103A BASIC/3000 Compiler provides a means of converting your BASIC programs, which you have previously debugged and stored via the BASIC/3000 Interpreter, into compiled form. This means that your BASIC programs will exist in the system as actual code segments, rather than as data in a data file. The result is that the programs may be run under the operating system, rather than through line-by-line interpreting via the Interpreter subsystem, with a consequent reduction of execution time.

Typically, compiled BASIC programs will execute many times faster than when run interpretively. Of course, it should be understood that actual speed improvement depends on the type of program and the resource demands on the MPE/3000 operating system. CPU bound programs, for example, will realize a speed improvement on the order of 10 to 30 times faster, with certain extraordinary routines running up to 100 times faster. I/O bound programs will run 1 to 4 times faster.

1-2. OVERVIEW OF OPERATION

Figure 1-1 illustrates the four major phases involved in compiling and using BASIC/3000 compiled programs. Detailed examples will be shown later, in Section III.

1-3. PROGRAM DEVELOPMENT The first phase is to write and debug BASIC/3000 programs in the usual way, via Interpreter commands and statements. The Interpreter, which operates as a subsystem under MPE/3000, constructs the interpretive version of your programs in files on disc. When you are satisfied that the programs execute properly in interpretive form, you then save the programs using the Interpreter's SAVE FAST command. The resulting file, called a BASIC fastsave file, is the only acceptable source input for the compiler.

1-4. COMPILE. The second phase is to compile the debugged interpretive programs which now reside in one or more BASIC fastsave files. The BASIC/3000 Compiler, operating as a subsystem under MPE/3000, compiles the programs into relocatable binary modules in a USL (User Subprogram Library) file when you give the COMPILE command.

1-5. PREPARE. The third phase is to prepare the USL file into runnable form. This is done, typically, by the MPE/3000 command :PREP. This command invokes the MPE/3000 Segmenter, which binds the relocatable binary modules into code segments in a program file.

1-6. EXECUTE. The fourth phase is to execute the program file, typically by the MPE/3000 command :RUN. During execution, reference will be made to subroutines in the Compiler Library, and also to a BASIC runtime library which exists in the system SL (Segmented Library).

Note: The "Prepare" and "Execute" phases can be accomplished automatically following compilation by use of the :BASICPREP and :BASICGO commands; see definitions in Section II.

1-7. LANGUAGE CONSIDERATIONS

There are a few semantic differences between interpretive and compiled BASIC that you should be aware of when writing BASIC programs that you intend to compile. These are described below.

1-8. INITIALIZING VARIABLES

Numeric variables are not preinitialized to the undefined number (1 in bit 0, followed by all zeros) unless CONTROL INIT is specified; see command definition in Section II. As in the Interpreter, the UND function returns true if variables have the undefined value. Strings will as usual be set to null by having their logical length set to zero.

1-9. NON-BASIC CALLS

The CALL statement may reference non-BASIC procedures in your USL file as well as those in a segmented library (SL).

1-10. NUMBER OF VARIABLES

The maximum number of program variables is limited to 182. Local variables within functions are limited to 120 per function (including any program variables referenced within the function). Common variables are limited to 239.

1-11. PROGRAM SIZE

The maximum code segment size on the HP 3000 Computer is 16K words. (K = 1024.) Normal system configurations limit the segment to 4K words. Since the compiled code generated is about 1 to 1-1/3 times the size of the interpretive program, the total size of the BASIC programs to be put into a single segment should be less than 3000 words (as determined with the Interpreter's LENGTH command). Programs larger than that should be modularized (using CHAIN and INVOKE) and placed into separate segments using the SEGMENT option of the Compiler's CONTROL command.

1-12. PROGRAM NAMES

Each subprogram is compiled as a Reclocatable Binary Module (RBM) in the USL. The name associated with that RBM is the filename of the BASIC fastsave program file, omitting any qualifying lockword and group and account names. Consequently, all BASIC programs to be compiled into the same USL file must have unique local filenames.

1-13. CHAIN AND INVOKE

You will have to keep track of any optional starting statement numbers, as these must be supplied to the compiler via the COMPILE command. Also, if you specify a program by means of a string expression (e.g., CHAIN A$) instead of by a literal string (e.g., CHAIN "PROGA"), you should keep track of all possible names that the expression can assume; these names should be supplied to the compiler via the ENTRY command.

1-14. COMMON BLOCKS

Even though your BASIC programs will exist in compiled form, there can be no correspondence between BASIC and FORTRAN COMMON areas. This is because the data structures are not compatible. FORTRAN uses fixed length strings; BASIC uses variable length strings. BASIC permits redimensioning of strings; FORTRAN does not.

1-15. FOR STATEMENT

If your program requires branching from one FOR-loop to another which uses the same FOR-NEXT variable name, you should be aware that the compiler will bind the NEXT with its static FOR (nearest preceding matching FOR). This is unlike the dynamic binding (remembers where it came from) provided by the Interpreter. See Figure 1-2 for comparison.

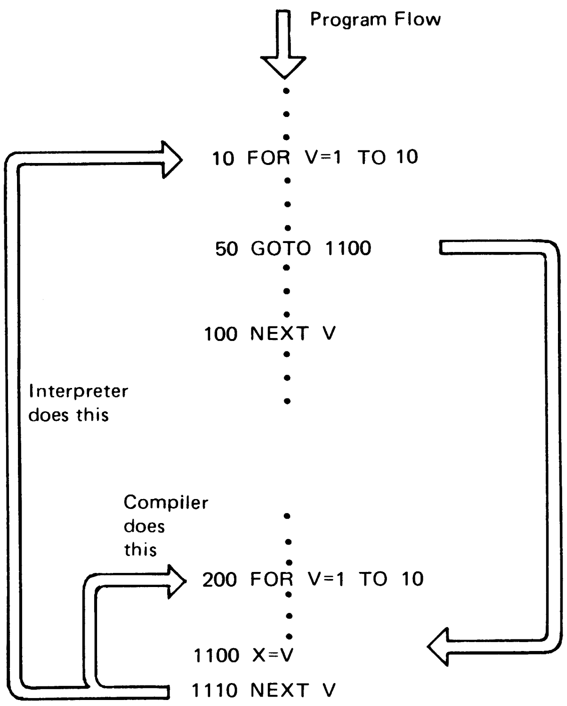


Figure 1-2. Branching into FOR Loop

1-16. ERROR CHECKING

It is assumed that any program which is to be compiled has been debugged previously. Thus there is little run-time error checking, which would slow down execution. Some of the interpretive error checking NOT done by the compiler:

a. subscript bounds checking

b. use of undefined variables

c. checking COMmon compatibilities

d. redimensioning past physical bounds.

1-17. NESTED BASIC

The use of a >BASIC command during an INPUT, LINPUT, or ENTER request is permitted only if the program is prepared with process-handling (PH) capability.

COMPILER COMMANDS

SECTION

II

Associated with the BASIC/3000 Compiler are "system" commands and "subsystem" commands. You use the system commands, which are recognized by MPE/3000, to access the compiler subsystem. Then, once the compiler is accessed, you use the subsystem commands to direct compilation of your BASIC programs. After exiting from the compiler subsystem, the system commands control the preparation and execution of the compiled programs.

2-1. SYSTEM COMMANDS

Three of the system commands defined below invoke the BASIC/3000 Compiler. These are :BASICOMP, :BASICPREP, and :BASICGO. The remaining three (:RUN, :PREPRUN, and :PREP) are associated only with preparing and running compiled programs.

The attributes of compiler files are listed in Table 2-1. Run-time input and list file attributes are listed in Table 2-2.

2-2. COMPILE ONLY

The :BASICOMP command invokes the compiler for compilation only. That is, on exiting from the compiler subsystem, the compiled programs are left in a USL file; preparation and execution do not follow automatically. The :BASICOMP command is written as follows:

```
:BASICOMP [commandfile] [,[uslfile] [,listfile]]
```

None of the file designators is necessary as they all have defaults associated with them. Any one or two, or all three of them can be present. The file designators are defined as follows:

commandfile
the file from which all subsystem commands are read; default is $STDIN. (Formal file designator is BSCTEXT.)

uslfile
the USL file on which the compiled object program is written. If omitted, $OLDPASS is used if it is an existing USL file; if not, $NEWPASS is used. (Formal file designator is BSCUSL.)

listfile
the file which receives all compiler list output. If omitted, $STDLIST is used. (Formal file designator is BSCLIST.)

2-3. COMPILE AND PREPARE

The :BASICPREP command invokes the compiler for compilation and the Segmenter for preparation. That is, on exiting from the compiler subsystem, preparation of the USL file into an executable program file occurs automatically. (The USL file is left in $OLDPASS unless $NEWPASS was used for the program file.) The :BASICPREP command is written as follows:

```
:BASICPREP [commandfile] [,[progfile] [,listfile]]
```

The file designators commandfile and listfile are as defined above. The progfile designator is defined as follows:

progfile
the program file onto which the prepared program segments are to be written. If omitted, the default file $NEWPASS is assigned. (Formal file designator is BSCPROG.)

2-4. COMPILE, PREPARE, AND EXECUTE

The :BASICGO command invokes the compiler for compilation and the Segmenter for preparation, and initiates program execution. That is, on exiting from the compiler subsystem, preparation and execution of the program file occurs automatically. (The program file is left in $OLDPASS.) The :BASICGO command is written as follows:

```
:BASICGO [commandfile] [,listfile]
```

2-5. PREPARE AND EXECUTE ONLY

The :RUN command executes a compiled BASIC program, assuming that it exists as a prepared program file. The :RUN command (in simplified form for most BASIC applications) is written as follows:

```
:RUN progfile [,NOECHO] [;PARM=label]
```

where

progfile
the name of the program file that contains the prepared program.

NOECHO
secondary entry point; indicates that input is not to be echoed to the output file. This parameter is effective only when input is not echoed automatically by the input device.

Table 2-1. Attributes of Compiler Files

	Command File	List File	USL File
Formal Designator			
User Specified	BSCTEXT	BSCLIST	BSCUSL
Default	$STDIN	$STDLIST	$OLDPASS/$NEWPASS
File Options			
Domain	OLD	NEW	OLD first NEW if old fails
Type	ASCII	ASCII	Binary
Record Format	Undefined	Variable	Fixed
Others	*Disallow :FILE	*Disallow :FILE Carriage control	*Disallow :FILE
Access Options			
Access	Read only	Write only	Input/output
Restriction	Share	Exclusive	Exclusive
Record Size	Default	Default	128
Device	Default	Default	Default
File Size	Default	5000	Default

* (if not user specified)

label
the statement number in the "main" BASIC program at which execution is to begin. This label must have been declared at compile time as an optional entry label; see COMPILE command.

Refer to the MPE/3000 reference manual if any of the other :RUN parameters is required.

The MPE/3000 commands :PREP and :PREPRUN may also be used for program preparation as described in the MPE/3000 reference manual. The optional NOECHO and PARAM= parameters are available following the uslfile parameter.

2-6. MAXDATA PARAMETER

If the executing BASIC program contains either FILES or COM, the MAXDATA parameter must be provided in the :PREP, :PREPRUN, or :RUN command. This is because the file buffers and common are dynamically allocated on the stack in the DL-to-DB space. MAXDATA must be specified large enough to accommodate the sum of the following:

a. File buffer space. This is the maximum sum of the lengths of the file buffers for all files open simultaneously. The length of a file buffer is the size of one record of the file.

b. Common space. This is the maximum sum of all COM areas active simultaneously.

c. Primary and secondary DB. 600 words maximum.

d. Program space. This is the amount of stack necessary for local variables. Summing of the space requirements for nested programs (INVOKE or multiline function reference) should be included. Program space can be taken from the STACK estimate on the label map.

Table 2-2. Attributes of Run-Time Input and List Files

	Input File	List File
Formal Designator		
User Specified	BASIN	BASLIST
Default	$STDINX	$STDLIST
File Options		
Domain	Old and Temporary	Old and Temporary
Type	ASCII	ASCII
Record Format	Variable	Variable
Others	Allow :FILE	Allow :FILE Carriage control
Access Options		
Access	Input only Non-multirecord access	Output only Non-multirecord access
Restriction	Default	Default
Record Size	-132	Default
Device	Default	Default
File Size	Default	1600

2-7. SUBSYSTEM COMMANDS

2-8. PROMPTS, COMMENTS, CONTINUATION

Examples:

```
$CONTROL<<CONTINUE COMMAND &
$ON NEXT LINE.>>SOURCE
```

The prompt character for compiler commands is the dollar sign ($). It may optionally be placed as the first character of each command in batch or session mode. Comments may be inserted before, after, or between command parameters. A comment is any string delimited by double less-than symbol at the start (<<) and double greater-than at the end (>>). To continue any command to the next line, use an ampersand (&) as the last character on the current line. Command records must be contained in columns 1 through 72.

2-9. CONTROL COMMAND

Example:

```
$CONTROL SOURCE,USLINIT,START=PROGA
```

The CONTROL command is used to set various compile options. It is not required if default or existing options are satisfactory for succeeding COMPILE commands. When used, you specify one or more of the following parameters, in any order, separated by commas. (Default conditions are underscored.)

SOURCE or NOSOURCE

The SOURCE parameter turns on the "list source" condition, requesting that the compiler list the programs to be compiled. Initially, the condition is off (no list). Once on, the condition remains on until turned off by NOSOURCE.

LABEL or NOLABEL

The LABEL parameter turns on the "label map" condition, requesting the compiler to print a label map showing the beginning octal code address for each statement. Initially, the condition is off (no map). Once on, the condition remains on until turned off by NOLABEL.

MAP or NOMAP

The MAP parameter turns on the "symbol table map" condition, requesting the compiler to print information regarding the structure, type, and octal address of each program variable for each program. Initially, the condition is off (no map). Once on, the condition remains on until turned off by NOMAP.

CODE or NOCODE

The CODE parameter turns on the "list code" condition, requesting the compiler to list (in octal) the object code generated for each program. Initially, the condition is off (no code list). Once on, the condition remains on until turned off by NOCODE.

LIST or NOLIST

The LIST parameter turns on the "list enable" condition, which must be on for the SOURCE, LABEL, MAP and CODE options to take effect. Initially, the condition is on (lists enabled). Once on, the condition remains on until turned off by NOLIST. When the condition is off, only error and warning messages and completion messages are sent to the listfile.

WARN or NOWARN

The WARN parameter turns on the "warning messages" condition, requesting the compiler to print warning messages about dubious though possibly acceptable language or command constructs. Initially, the condition is on (warnings printed). The condition remains on until turned off by NOWARN.

USLINIT

This parameter initializes the uslfile to "empty" status. This is necessary when you are compiling to an old USL file and you want to eliminate the current contents. This action is automatically performed by the compiler when the uslfile is currently empty or has invalid contents.

SEGMENT=segname

This parameter sets the segment name to any specified name acceptable to the MPE/3000 Segmenter (i.e., starts with alphabetic, up to 15 characters in length, and may contain alphanumeric and apostrophe characters). All subsequent programs compiled will be put in that segment until a new segment is specified by SEGMENT=. This action is necessary especially when program units being compiled into one segment approach the segment size limit. (The default segment is SEG'.)

START=progname

This parameter specifies which BASIC program (in a CHAIN/INVOKE-connected series) is the main program; i.e., which one is to receive initial control when prepared and run. The default starting program is the first program compiled. This parameter is not accepted when the "subprogram" condition is on; otherwise it can occur anywhere and the last occurence will hold.

SUBPROGRAM

This parameter turns on the "subprogram" condition, which inhibits the normally automatic generation of an outer block in the compiled code. Initially, the condition is off (outerblock automatically generated). Once turned on, this condition cannot be disabled. SUBPROGRAM is mutually exclusive with START and ENTRY during one invocation of the compiler.

LINES=number

This parameter sets the page size of the listfile to any specified number from 10 to 9999 lines per page. Default is 60. Whenever the next line sent to the listfile would overflow the page size, the compiler precedes it with a page eject, a page heading, and two blank lines.

INIT

This parameter inserts initializing code into the programs which will be compiled by the next COMPILE statement; this code initializes all local numeric simple variables and arrays to the "undefined" number (1 in bit 0, followed by all zeros). This value may be checked for with the UND function. INIT is enabled for only one COMPILE command that requires initialization code.

2-10. ENTRY COMMAND

Example:

```
$ENTRY SUBPROG1, SUBPROGA,A
```

The ENTRY command is used to inform the compiler of all possible program names that may be assumed by string expressions in CHAIN or INVOKE statements in your main program and subprograms. Thus if there is an INVOKE A$ in a source program, all values which could be assumed by A$ should be specified in an ENTRY command; it is not necessary if CHAIN or INVOKE specifies a literal string (e.g., INVOKE “A”). Any number of ENTRY commands may be given; each one simply adds more names to the chain-invoke table. (A reference that is not in the table is treated as a library call. In such a case, your program executes a dynamic LOADPROC from the segmented libraries; this is costly in execution time.) The ENTRY command cannot be used with a SUBPROGRAM compilation.

2-11. TITLE COMMAND

Example:

```
$TITLE “PROGRAM ”“C”“ LISTING”
```

The TITLE command replaces the title portion (columns 29 through 132) of the standard page heading with the specified string or strings. The title begins in column 29 and is extended by blanks if less than 104 characters. If more than one string is specified, the separating commas, blanks, or comments are not included in the title, and the strings are concatenated. Two adjacent quotes can be used to include one quote within the string. If no parameter is present, the title portion will be blank.

2-12. COMPILE COMMAND

Example:

```
$COMPILE PROG1, PROG2 (100, 250)
```

The COMPILE command causes compilation of the specified BASIC fastsave programs to begin. If a source program has multiple entry points, each entry point (other than the starting statement) must be listed in parentheses following the program name. Programs in other accounts and/or groups may be compiled into your USL file by using a fully qualified file designator, of the form:

```
filename [/lockword] [.group[.account]]
```

For example:

```
$COMPILE PROGX.PUB.SYS
```

However, only filename is associated with the RBM in the USL file.

If an active entry with the same name already exists in the USL file, the existing entry is made inactive. This ensures that the most recently compiled program is the one prepared into the program file.

Any desired compile options must be established with a CONTROL command prior to issuing COMPILE.

2-13. EXIT COMMAND

Example:

```
$EXIT
```

The EXIT command terminates use of the BASIC/3000 Compiler.

PROGRAMMING EXAMPLES

SECTION III

Figures 3-1 and 3-2 illustrate typical examples of using the BASIC/3000 Compiler. For both examples, it is assumed that the BASIC programs have been entered into the system via the BASIC/3000 Interpreter, have been debugged, and have been SAVEd with the FAST option.

In the printouts shown in the examples, user inputs are distinguished by system prompts (:) or subsystem prompts ($). All other printouts are system or subsystem responses.

3-1. SINGLE-PROGRAM EXAMPLE

Figure 3-1 shows the compilation, preparation, and execution of a simple 12-line BASIC program, the purpose of which is to list the contents of any ASCII file. The name of the program is FILELIST.

The step-by-step commentary facing the printout explains the actions occurring as a result of each input command.

3-2. SEGMENTED PROGRAMS EXAMPLE

Figure 3-2 shows the compilation, preparation and execution of programs which are linked by CHAIN and INVOKE statements and are compiled into several segments.

Program A is the main program. When run, this program will receive initial control from the outer block. It will prompt a user for a program name (B or C) to execute next. Thus program A will selectively INVOKE to either program B or program C. Program B also chains to program D.

As described in Figure 3-2, programs B and D are compiled into SEG3, C into SEG2, and A into SEG1.

To illustrate the actions occurring when these segmented programs are executed, consider a response of "B" to the program prompt, "PROGRAM TO EXECUTE?" In such a case, program control is as follows: when A receives control, it invokes B. B then chains to D. When D executes an END statement, control passes back to A. A terminates by executing a STOP statement.

A time sequence showing the run-time environment is given in Figure 3-3. The sequence shows the code segment and stack conditions at five points in time. Operation is as follows.

TIME 1

The outer block, B'RUNOB, receives initial control from the loader. The outer block calls a library procedure which opens the following files.

BASIN
the input file, which is read from by the statements INPUT, ENTER, and LINPUT. The default for this file is $STDINX. It can be overridden using the formal designator BASIN in an MPE/3000 :FILE command.

BASLIST
The output file, which is written to by the statements PRINT, PRINT USING, MAT PRINT, and MAT PRINT USING. The default for this file is $STDLIST. It can be overridden using the formal designator BASLIST in an MPE/3000 :FILE command.

Once the files are open, buffer space is allocated dynamically on the stack. Other BASIC run-time variables are also initialized in primary and secondary DB. Then the program specified by START= (program A) during the compilation is invoked.

TIME 2

Program A, in segment SEG1, receives control. It allocates its program variables on the stack and then asks for a program name to be input. The name read in was B, so program A invokes to program B.

TIME 3

Program B, in segment SEG3, receives control and does local allocation of program variables. COMmon space is also allocated by B and it passes information to D through that COMmon. The actual COMmon space is dynamically allocated in the DL to DB area and is completely managed for the user. Pointers to the COMmon space are in primary DB. Program B now chains to D.

TIME 4

The program local space for program B is overlayed by the local space of program D. The space does not need to be saved because a CHAIN does not return. Program D, also in segment SEG1 (but not required to be) receives control and allocates its local variables. When it executes an END statement, program D exits back to program A. COMmon is deallocated prior to the exit.

TIME 5

Program A, in segment SEG1, receives control again and requests the next program to execute. Because a null string (zero characters) is input, program A terminates in a STOP statement.

After logging on (or following a :JOB card in batch), specify the file characteristics for the USL file which is to receive the compiled code, namely: it is a NEW file to be SAVEd for future access. For this example, the USL file will be named USLF. Next, load the compiler via the BASICOMP command. This command opens three files. Defaults are accepted for commandfile (note comma) and for listfile. These defaults are $STDIN and $STDLIST respectively. For uslfile, a back reference is made to USLF (note asterisk). ❶

In response to compiler prompts, specify desired compile options via CONTROL commands: initialize the uslfile (USLINIT) and request a SOURCE listing, a symbol MAP, and a LABEL map. The compiled code will be put into a segment called TESTSEG. Then give the COMPILE command. ❷

The compiler now prints out a source listing and compiles the program. ❸

Next the symbol map is printed, showing where each variable is located in the stack. The ",I" following an address indicates an indirect address (see string variables). Addresses are in octal. ❹

Next the label map is printed, giving the starting location in the code list for each statement. (CODE list was not requested in this example.) Statements 10, 20, 100, and 110 are shown "equivalent" to other statements. This indicates, for example, that any branch to line 110 will actually branch to line 80. Lines 120 and 130 are "equivalent" to the end of the program. The ENTRY POINT is the octal address at which execution will begin. STACK indicates how much stack is required for this program; this can be used in estimating the MAXDATA parameter when preparing the program file (see "MAXDATA Parameter" in Section II). ❺

Compilation is now complete. In response to the next command prompt, exit back to MPE/3000. ❻

Now prepare the USL file into a job-temporary program file named BASPROG. MAXDATA must be specified; 2000 words is estimated for this example. A PMAP is requested, though it is not necessary. ❼

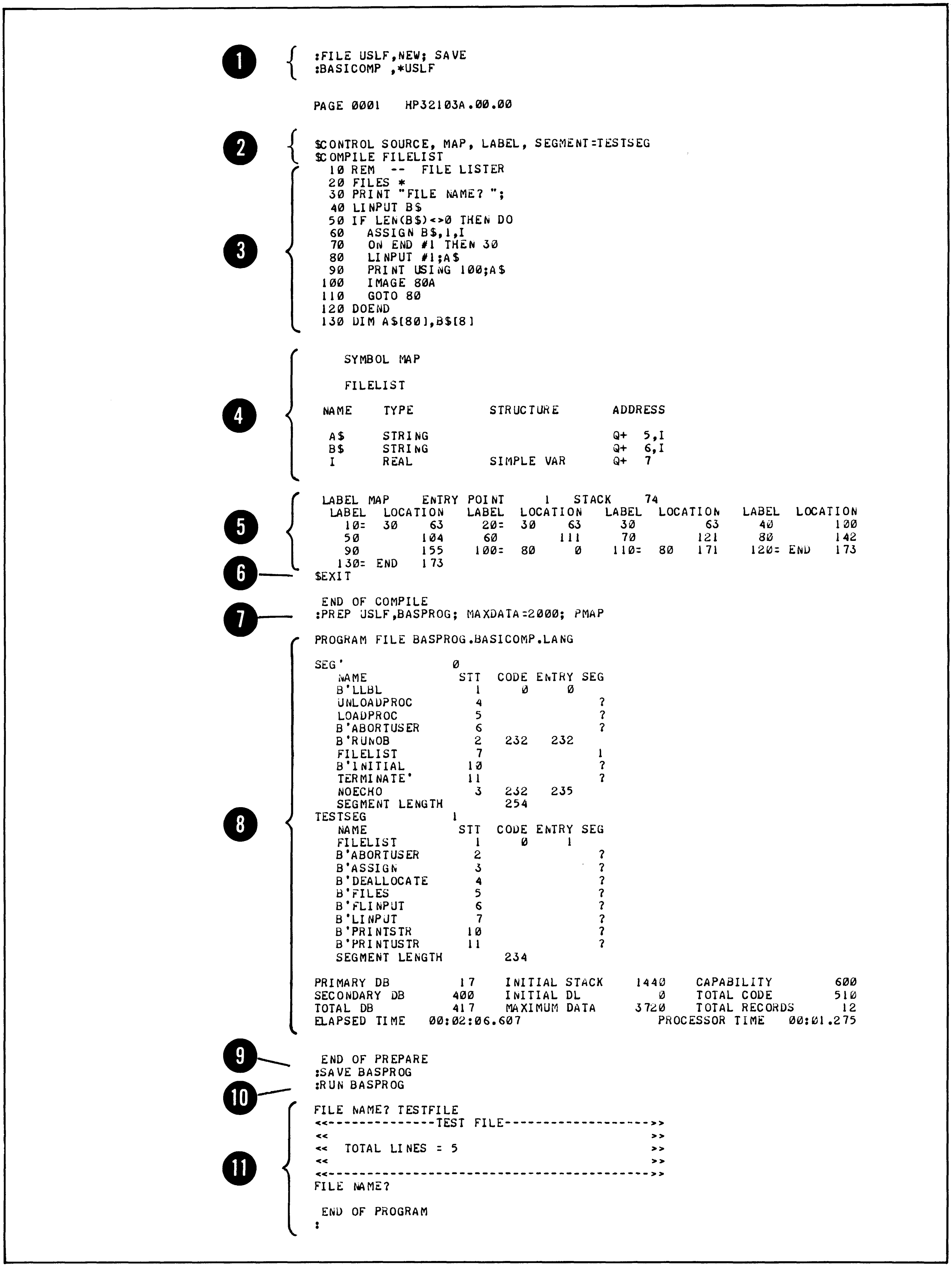
After the file is prepared, MPE/3000 prints out the PMAP. Note that two segments were prepared: the default segment SEG' (which contains an outer block generated by the compiler with an external reference to the compiled program, FILELIST) and the TESTSEG segment which contains the program. ❽

To make the job-temporary program file BASPROG permanent, specify a SAVE command. ❾

Now that the program exists as a program file, it can be run. ❿

The compiled BASIC program runs, prints out a prompt for input. We respond with a file name which, for illustrative purposes here, is a test file called TESTFILE consisting of 5 records. We terminate by typing a carriage return in response to the next prompt. MPE/3000 prints an END OF PROGRAM message and prompts for the next command. ⓫

Figure 3-1A. Compilation of Single Program



```
:FILE USLF,NEW; SAVE
:BASICOMP ,*USLF

PAGE 0001   HP32103A.00.00

$CONTROL SOURCE, MAP, LABEL, SEGMENT=TESTSEG
$COMPILE FILELIST
  10 REM  --  FILE LISTER
  20 FILES *
  30 PRINT "FILE NAME? ";
  40 LINPUT B$
  50 IF LEN(B$)<>0 THEN DO
  60   ASSIGN B$,1,I
  70   ON END #1 THEN 30
  80   LINPUT #1;A$
  90   PRINT USING 100;A$
 100   IMAGE 80A
 110   GOTO 80
 120 DOEND
 130 DIM A$[80],B$[8]

    SYMBOL MAP

    FILELIST

 NAME    TYPE          STRUCTURE       ADDRESS

  A$    STRING                         Q+  5,I
  B$    STRING                         Q+  6,I
  I     REAL          SIMPLE VAR       Q+  7

 LABEL MAP   ENTRY POINT    1   STACK   74
  LABEL LOCATION   LABEL LOCATION  LABEL  LOCATION  LABEL LOCATION
    10= 30    63    20=  30    63    30         63    40       100
    50       104     60       111    70        121    80       142
    90       155   100=  80     0   110=  80   171   120= END  173
   130= END  173
$EXIT

 END OF COMPILE
:PREP USLF,BASPROG; MAXDATA=2000; PMAP

PROGRAM FILE BASPROG.BASICOMP.LANG

SEG'              0
   NAME            STT  CODE ENTRY SEG
   B'LLBL            1     0     0
   UNLOADPROC        4               ?
   LOADPROC          5               ?
   B'ABORTUSER       6               ?
   B'RUNOB           2   232   232
   FILELIST          7               1
   B'INITIAL        10               ?
   TERMINATE'       11               ?
   NOECHO            3   232   235
   SEGMENT LENGTH        254
TESTSEG           1
   NAME            STT  CODE ENTRY SEG
   FILELIST          1     0     1
   B'ABORTUSER       2               ?
   B'ASSIGN          3               ?
   B'DEALLOCATE      4               ?
   B'FILES           5               ?
   B'FLINPUT         6               ?
   B'LINPUT          7               ?
   B'PRINTSTR       10               ?
   B'PRINTUSTR      11               ?
   SEGMENT LENGTH        234

PRIMARY DB         17    INITIAL STACK    1440    CAPABILITY        600
SECONDARY DB      400    INITIAL DL          0    TOTAL CODE        510
TOTAL DB          417    MAXIMUM DATA     3720    TOTAL RECORDS      12
ELAPSED TIME   00:02:06.607                  PROCESSOR TIME   00:01.275

 END OF PREPARE
:SAVE BASPROG
:RUN BASPROG

FILE NAME? TESTFILE
<<--------------TEST FILE-------------------->>
<<                                           >>
<<  TOTAL LINES = 5                          >>
<<                                           >>
<<------------------------------------------->>
FILE NAME?

 END OF PROGRAM
:
```

Figure 3-1B. Compilation of Single Program

Defaults are accepted for uslfile ($NEWPASS) and listfile ($STDLIST). Commandfile is TEXT. ❶

Page title. ❷

Compile programs B and D into segment SEG3. Program B has optional entry points at statements 50 and 100. Also, program B may be called by a string expression, so its name is entered in the Chain-Invoke table. ❸

Compile program C into SEG2 with optional entry points at statements 50 and 100. Put name C in Chain-Invoke table. ❹

Compile program A into SEG1. Declare this to be the main program. Request source listing. ❺

Source listing of main program. ❻

Exit to MPE/3000. ❼

Request to prepare and run programs in uslfile (now $OLDPASS) and provide a PMAP. ❽

PMAP begins. ❾

Outer block in SEG' contains STT links to programs, B, C, and A, in segments 3, 2, and 1 respectively. ❿

Segment 1 (SEG1) contains program A. ⓫

Segment 2 (SEG2) contains program C. ⓬

Segment 3 (SEG3) contains programs D and B. ⓭

End of PMAP. ⓮

Sample run of program. Terminates by input of null string. ⓯

Figure 3-2A. Compilation of Segmented Programs

```
①    :BASICOMP TEXT

②    PAGE 0001   HP32103A.00.00

     $CONTROL USLINIT,SEGMENT=SEG3
③    $COMPILE B(50,100),D
     $ENTRY B
     $CONTROL SEGMENT=SEG2
④    $COMPILE C(50,100)
     $ENTRY C
     $CONTROL SEGMENT=SEG1
⑤    $CONTROL SOURCE,START=A
     $COMPILE A
       10 DIM P$[8]
       20 INPUT "PROGRAM TO EXECUTE?",P$
       30 IF LEN(P$)<>0 THEN DO
⑥      40   INVOKE P$,50
       50   GOTO 20
       60 DOEND
       70 STOP
⑦    $EXIT

      END OF COMPILE
⑧    :PREPRUN $OLDPASS;PMAP

⑨    PROGRAM FILE $NEWPASS.BASICOMP.LANG

     COMMON ARRAY ALLOCATION

        NAME                 ADR   LEN
        COM0'                  ?     ?

     SEG'             0
        NAME             STT  CODE ENTRY SEG
        B'LLBL            1     0     0
        UNLOADPROC        3                ?
        LOADPROC          4                ?
        B'ABORTUSER       5                ?
        B                 6                3
⑩      C                 7                2
        B'RUNOB           2   230   230
        A                10                1
        B'INITIAL        11                ?
        TERMINATE'       12                ?
        SEGMENT LENGTH        250
     SEG1             1
        NAME             STT  CODE ENTRY SEG
⑪      A                 1     0     0
        B'ABORTUSER       2                ?
        B'INPUTSTR        3                ?
        B'STOP            4                ?
        B'LLBL            5                0
        SEGMENT LENGTH        130
     SEG2             2
        NAME             STT  CODE ENTRY SEG
⑫      C                 1     0     0
        B'ABORTUSER       2                ?
        B'PRINTSTR        3                ?
        SEGMENT LENGTH        110
     SEG3             3
        NAME             STT  CODE ENTRY SEG
        D                 1     0     0
        B'ABORTUSER       3                ?
⑬      B'DEALLOCATE      4                ?
        B'ALLOCATE        5                ?
        B'PRINTSTR        6                ?
        B                 2   103   103
        SEGMENT LENGTH        220

     PRIMARY DB        20    INITIAL STACK     1440    CAPABILITY          600
     SECONDARY DB     377    INITIAL DL           0    TOTAL CODE          730
     TOTAL DB         417    MAXIMUM DATA         ?    TOTAL RECORDS        14
⑭    ELAPSED TIME   00:02:52.405                     PROCESSOR TIME   00:01.746

      END OF PREPARE

⑮    PROGRAM TO EXECUTE?B
     PROGRAM TO EXECUTE?""

      END OF PROGRAM
     :
```

Figure 3-2B. Compilation of Segmented Programs

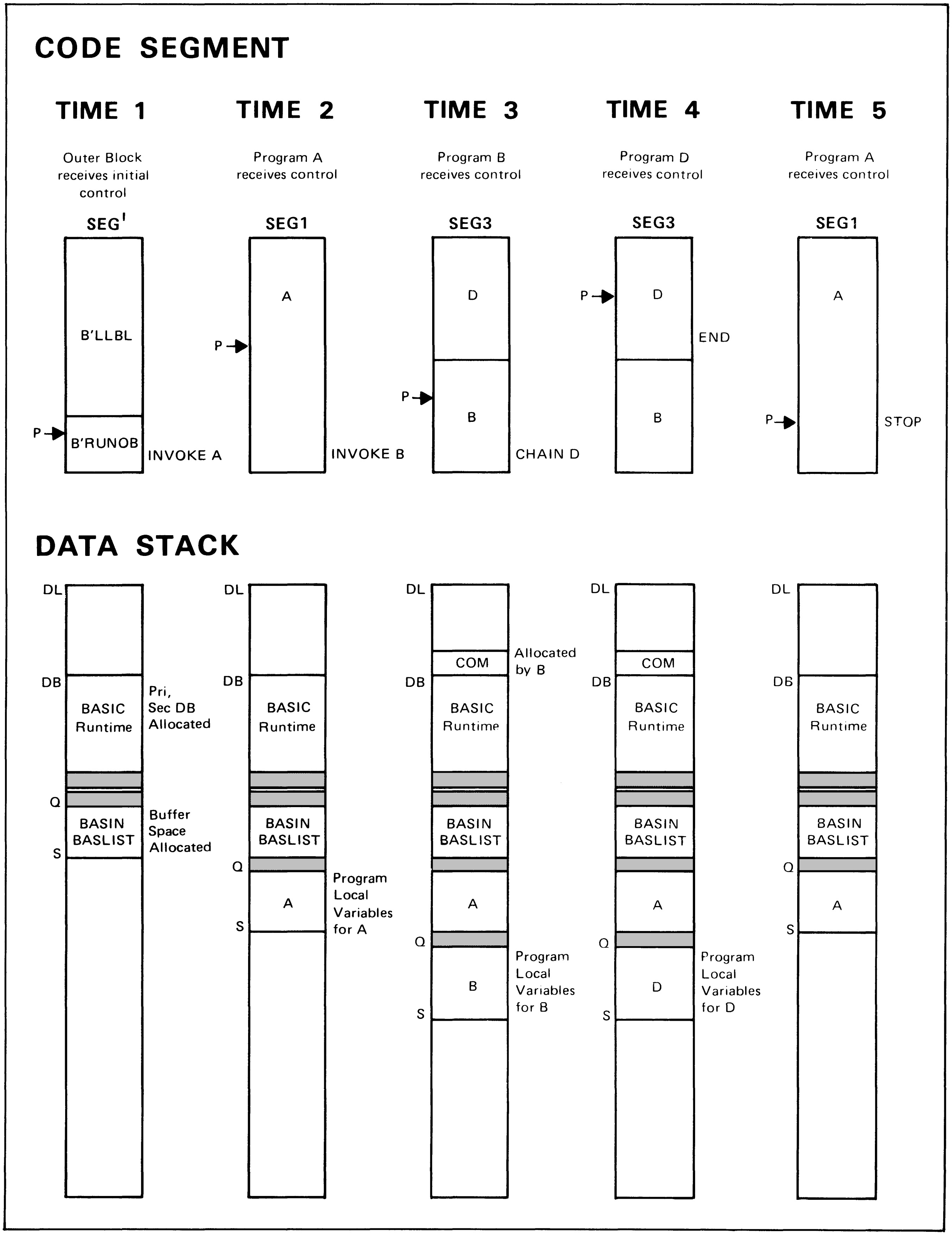


Figure 3-3. Run-Time Environment of Segmented Example

INTERPRETING MESSAGES

SECTION IV

4-1. COMPILE-TIME ERROR MESSAGES

Error messages which may occur during the compile phase are presented as a WARNING or ERROR in the following format:

```
**WARNING
                message [: addendum]
**ERROR
```

where the message is one of those listed below, and the addendum is additional information given with some error messages to further identify the cause of the error.

1 COMPILER OUT OF SPACE
Space requirements for compiling program exceed the available memory. See your system manager. The compiler must be prepared (:PREP) with a larger MAXDATA, and/or the maximum stack segment size (defined during system configuration) must be expanded.

2 INVALID COMMAND
Not a valid compiler command.

3 INVALID CHARACTER
Extraneous character in command processing.

4 USL FILE PREVIOUSLY INITIALIZED
Given when USLINIT parameter is specified but USL file has already been initialized.

5 USLINIT INVALID AFTER $COMPILE COMMAND
Initialization of the USL file is invalid after a program has been compiled.

6 INVALID NUMBER OF LINES
Number of lines cannot be set less than 10 or greater than 9999.

7 EXPECTED A QUOTE
TITLE string must be delimited by a quote.

8 START INVALID UNDER SUBPROGRAM MODE
When SUBPROGRAM is specified, all subsequent START parameters are ignored.

9 EXPECTED PROGRAM NAME
A program name is expected as the parameter to COMPILE command.

10 EXPECTED A <
Comments must begin with<<; given if only a single< is detected.

11 EXPECTED A &
Commands which are continued must have ampersand as last character of the line; given if comma is last character of the line.

12 EXPECTED A =
LINES, START, and SEGMENT all must be followed by an equal sign.

13 INVALID SUBPARAMETER
Subparameter must be a name for START= and SEGMENT=.

14 SUBPARAMETER NAME TOO LONG, TRUNCATED TO 15 CHARACTERS

15 $CONTROL PARAMETER EXPECTED

16 USL FILE INITIALIZED DUE TO BAD CHARACTERISTICS
Invalid directory or directory element. Given on COMPILE if initialization is required but user fails to specify USLINIT.

17 INVALID $CONTROL PARAMETER

18 $ENTRY INVALID UNDER SUBPROGRAM MODE
When SUBPROGRAM is specified, all subsequent ENTRY commands are ignored.

19 $TITLE STRING TOO LONG, TRUNCATED TO 104 CHARACTERS

20 EXTRANEOUS CHARACTER

21 UNABLE TO OPEN FASTSAVE FILE
The file may not exist or cannot be opened because of access or security restrictions.

22 FILE NOT TYPE FASTSAVE
File specified in COMPILE command is not a BASIC fastsave file.

23 I/O ERROR ON FASTSAVE FILE

24 BAD FASTSAVE FILE ATTRIBUTES
Recreate the fastsave file through the BASIC/3000 Interpreter.

26 SUBPROGRAM INVALID AFTER $ENTRY OR START
After ENTRY command or START parameter has been specified, SUBPROGRAM parameter is ignored.

27 TOO MANY PROGRAM VARIABLES
The number of program variables is limited; see paragraph 1-10 in Section I. Reduce number of variables by "reusing" variables, putting values into arrays, or putting variables in COMmon.

28 INVALID PROGRAM NAME
Program names for COMPILE command must be alphanumeric and begin with alphabetic.

29 QUALIFIER DELETED
Group and account names and lockword are deleted from program name and not used as part of the RBM symbolic name.

30 UNABLE TO PROCESS $ENTRY COMMANDS
A temporary file is necessary and could not be opened.

31 TOO MANY FUNCTION VARIABLES
Function references too many local and program variables; see paragraph 1-10 in Section I.

32 $ENTRY NAME TOO LONG, TRUNCATED TO 8 CHARACTERS
ENTRY names cannot exceed eight characters.

33 CODE SEGMENT > 16K WORDS
The program being compiled creates a code segment which is more than 16,384 words, the largest code segment size possible. The program should be divided into smaller programs.

34 MORE THAN 16 FILES
A maximum of 16 files is allowed.

35 USL FILE SPACE EXCEEDED
The space in the USL file was exceeded. Build the USL file with more records and recompile.

Messages 36 through 52 refer to PRINT USING strings specified as string expressions. The format string is echoed with a caret character or up-arrow (depending on particular device) indicating the point at which the error was detected.

36 INVALID CHARACTER
37 MISSING GROUP REPLICATOR
38 REPLICATOR NOT ALLOWED
39 MISSING RIGHT QUOTE
40 MISSING NUMBER AFTER APOSTROPHE
41 INVALID NUMERIC SPECIFICATION
42 NUMBER > 255
43 MISSING D-PART
44 REPLICATOR > 255
45 INVALID STRING SPECIFICATION
46 MISSING LEFT PARENTHESIS IN C-SPECIFICATION
47 MISSING SEPARATOR
48 INVALID C-SPECIFICATION
49 UNBALANCED PARENTHESES
50 MISSING RIGHT PARENTHESIS
51 INVALID COMPLEX SPECIFICATION
52 REPLICATOR = 0

63 COMMAND FILE RECORD TOO LONG, TRUNCATED TO 80 CHARACTERS

64 BAD LIST FILE CHARACTERISTICS
Listfile is not type ASCII, or record size is less than 40 characters, or the compiler could not get write access to listfile.

65 UNABLE TO OPEN LIST FILE

66 UNABLE TO OPEN COMMAND FILE

67 BAD COMMAND FILE CHARACTERISTICS
Commandfile is not type ASCII or is the $NULL file, or the compiler could not get read access to the commandfile.

68 UNABLE TO OPEN USL FILE

69 BAD USL FILE CHARACTERISTICS
USL file is not binary, or the file is not type USL, or the record size is not 128 words, or the compiler could not get read and write access to the file.

70 END OF FILE ON COMMAND FILE
The EXIT command is missing or misplaced in the job stream.

71 END OF FILE ON USL FILE
The USL file is invalid. Recompile into a new USL file.

72 END OF FILE ON FASTSAVE FILE
The fastsave file is invalid. Recreate the file through the BASIC/3000 Interpreter.

73 END OF FILE ON TEMPORARY FILE
This is a compiler error. See your system manager.

74 END OF FILE ON LIST FILE

75 I/O ERROR ON COMMAND FILE

76 I/O ERROR ON USL FILE

77 I/O ERROR ON TEMPORARY FILE

78 I/O ERROR ON LIST FILE

79 I/O ERROR ON UNKNOWN FILE
This is a compiler error. See your system manager.

80 UNABLE TO CLOSE COMMAND FILE

81 UNABLE TO CLOSE USL FILE

82 UNABLE TO CLOSE FASTSAVE FILE

83 UNABLE TO CLOSE LIST FILE

84 UNABLE TO CLOSE UNKNOWN FILE
This is a compiler error. See your system manager.

4-2. RUN-TIME WARNING MESSAGES

The execution of compiled BASIC programs is similar to running a program in the interpreter with the NOWARN option; i.e., most warning messages (such as overflow and underflow warnings) are suppressed. The few warnings given while running compiled programs allow for recovery during input when the input file is interactive with either the output file or $STDLIST.

BAD INPUT—RETYPE FROM ITEM nn

A string was input to a numeric variable or vice versa; or the form of a numeric item is incorrect.

STRING OVERFLOW—RETYPE FROM ITEM nn

The string item exceeds 255 characters.

>BASIC NOT ALLOWED

The >BASIC option was disabled with BRK function; the user is not allowed to use this feature.

TRANSMISSION ERROR—RE-INPUT LINE

A transmission error occured during input, causing characters to be lost.

4-3. RUN-TIME ERROR MESSAGES

When an error is detected during execution, one or more messages are printed to aid the user in isolating the problem. If the BASLIST file and $STDLIST files are different, the messages are printed on both files.

Run-time error messages include the following information:

a. A message of the form

**ERROR nn: message

where nn and message are as listed below.

b. For file errors, a file information display indicating the characteristics of the file and the exact nature of the file error.

c. A stack traceback indicating where the error occurred. Compiler library routines are denoted by prefix S; user library routines by P or G; program modules have no prefix. An example of a stack traceback is shown in Figure 4-1.

0 OUT OF DATA
Exhausted the DATA list while attempting to execute a READ or MAT READ statement.

1 NUMERIC DATA FOR STRING ITEM
The destination variable in a READ or MAT READ statement was string but the next item in the DATA list was numeric.

2 SUBSTRING DESIGNATOR EXCEEDS LOGICAL LENGTH PLUS ONE
The first substring designator for a destination string must always be less than or equal to string's logical length plus one to ensure a "contiguous" string.

4 DESTINATION SUBSTRING EXCEEDS PHYSICAL SIZE
The first substring designator for a destination string must be less than or equal to the physical length; and the substring must not extend beyond the physical length.

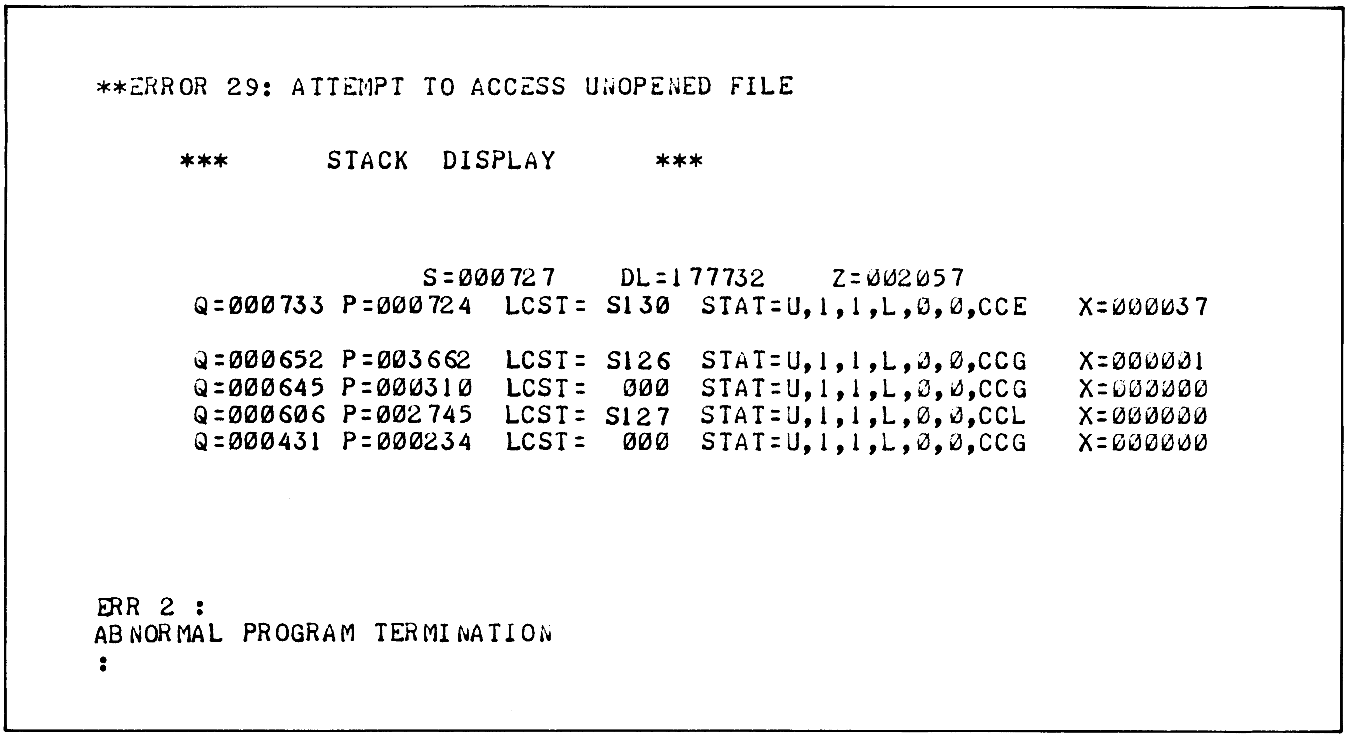
5 FIRST SUBSTRING DESIGNATOR < = ZERO

6 UNABLE TO FIND CHAIN/INVOKE PROCEDURE: name
The CHAIN/INVOKE procedure does not exist in the program file or in an SL file.

7 INVALID CHAIN/INVOKE PROCEDURE NAME
CHAIN/INVOKE name resulting from a string expression did not have the form of a file name:

filename [/lockword] [.group[.acct]]

Specifically, the name erroneously began with a non-alphabetic or contained non-alphanumeric characters other than a slash (/) or period, or slash or period was followed by non-alphabetic character.



```
**ERROR 29: ATTEMPT TO ACCESS UNOPENED FILE

      ***        STACK  DISPLAY        ***


                    S=000727    DL=177732     Z=002057
      Q=000733 P=000724  LCST= S130  STAT=U,1,1,L,0,0,CCE     X=000037

      Q=000652 P=003662  LCST= S126  STAT=U,1,1,L,0,0,CCG     X=000001
      Q=000645 P=000310  LCST=  000  STAT=U,1,1,L,0,0,CCG     X=000000
      Q=000606 P=002745  LCST= S127  STAT=U,1,1,L,0,0,CCL     X=000000
      Q=000431 P=000234  LCST=  000  STAT=U,1,1,L,0,0,CCG     X=000000




ERR 2 :
ABNORMAL PROGRAM TERMINATION
:
```

Figure 4-1. Stack Traceback Example

8 INTEGER OVERFLOW
Integer expression resulted in a value less than -32768 or greater than +32767.

9 INTEGER DIVIDE BY ZERO

10 LOG OF NEGATIVE NUMBER

11 SQR OF NEGATIVE NUMBER

12 ZERO TO ZERO POWER

13 ZERO TO INTEGER POWER< = ZERO

14 NEGATIVE NUMBER TO REAL POWER
Floating point base (type real or long) to a floating point power (results in type complex value).

15 STRING EXCEEDS 255 CHARACTERS
Maximum source string length in any expression is 255 characters.

16 INVALID SUBSTRING DESIGNATOR
For a string variable of the form
S$(f,l)
the second substring designator must not be less than the first minus one. For a string variable of the form
S$(f;l)
the second substring designator must be greater than or equal to zero.

17 FAILURE TO "CREATE" BASIC.PUB.SYS
The BASIC/3000 Interpreter does not exist in the system. See system manager.

18 FAILURE TO "ACTIVATE" BASIC.PUB.SYS
Cannot run the BASIC/3000 Interpreter. See system manager.

21 ">EOD" INPUT
End of input detected due to ">EOD".

23 INVALID STRING INPUT
Unbalanced quotes in string read by INPUT statement.

25 INVALID NUMBER INPUT
Expected numeric constant input but the constant had invalid form, or the destination variable is type integer and the constant is less than -32767 or greater than +32767.

26 INVALID FILE NUMBER
File number specified in a FILES statement is less than or equal to zero, or exceeds the number of local files.

27 NEGATIVE FILE SIZE
File limit specified in CREATE statement is negative.

28 INVALID RECORD SIZE
Record size specified in a CREATE statement is less than 4 or greater than 319.

29 ATTEMPT TO ACCESS UNOPENED FILE
Reference was made (e.g., in a READ# statement) to a file which is not currently open.

30 ATTEMPT TO LOCK NON-LOCKABLE FILE
Attempted to execute LOCK# for a file not opened with NL or WL option.

31 ATTEMPT TO LOCK SECOND FILE
Only one file may be locked (with LOCK#) at a time.

32 ATTEMPT TO UNLOCK NON-LOCKABLE FILE
Attempted to execute UNLOCK# for a file not opened with NL or WL option.

33 ADVANCE ATTEMPTED ON NON-BASIC FILE
ADVANCE# may be used only for BASIC formatted files.

34 UNABLE TO OPEN FILE: TOO MANY FILES
FILES or ASSIGN statement cannot open file because 16 files are currently open.

35 UNABLE TO OPEN USER FILE: filename
FILES statement cannot open file. A slash following filename indicates the presence of a password, which is not shown.

36 UPDATE ATTEMPTED ON NON-BASIC FILE
UPDATE# may be used only for BASIC formatted files.

37 UNABLE TO OPEN FILE: INSUFFICIENT SPACE FOR BUFFER
FILES or ASSIGN statement cannot open file. Use larger MAXDATA in :RUN or :PREP command.

38 INVALID LINKED-FILE NUMBER
Linked-file number (specified on FILES statement in the form "#integer") exceeds the number of global files.

39 INSUFFICIENT SPACE FOR COMMON
Unable to expand stack for COMmon area specified in COM statement. Use larger MAXDATA in :RUN or :PREP command.

40 LINPUT# ATTEMPTED ON NON-ASCII FILE
LINPUT# may be used only for ASCII files.

41 ATTEMPT TO WRITE ON READ-ONLY FILE

42 ATTEMPT TO READ FROM WRITE-ONLY FILE

43 DIRECT ACCESS ATTEMPTED ON SERIAL FILE

44 END OF FILE
End-of-file detected and no ON END# for that file was in effect.

45 NUMERIC UPDATE ITEM FOR STRING DATA
The expression in the UPDATE# statement was numeric but the next item in the BASIC formatted file was a string.

46 CHAIN/INVOKE/RUN TO UNDECLARED LABEL
Destination label specified in a CHAIN or INVOKE statement or the PARM= parameter in the :RUN command was not declared in a label list at compile time. Destination program must be recompiled.

47 STRING EVALUATES TO INVALID NUMBER IN CONVERT
For CONVERT of string to number, the string expression resulted in a number of invalid form, or the destination variable is type integer and the resulting number is less than -32767 or greater than +32767.

48 INCOMPATIBLE ARRAY DIMENSIONS
In a MAT statement, the destination array is one-dimensional but the right-hand side results in a two-dimensional with more than one column.

49 INVALID DAT$ PARAMETER
The first-character designator is less than 1 or exceeds the last-character designator. Or the last-character designator exceeds 27.

50 COLUMNS < > ROWS IN MAT MULTIPLY
For MAT multiplication, the multiplicand must have dimensions m-by-p while the multiplier must have dimensions p-by-n.

51 DESTINATION ARRAY NOT LARGE ENOUGH
In a MAT statement, the number of elements contained in the right-hand side exceeds the maximum number of elements in the destination array.

55 ATTEMPT TO EXECUTE FNEND
Missing function RETURN statement.

56 MISSING SUBSCRIPT IN FIRST OCCURRENCE OF COM
First occurrence of COM declaration uses (*) or (*,*) indicators. Dimensions must be declared explicity or must CHAIN/INVOKE from a program which declares COM explicitly.

Messages 57 through 75 all refer to a PRINT USING format specified as a string expression. The format string is echoed with a caret character or up-arrow (depending on particular device) indicating the point at which the error was detected.

57 INVALID CHARACTER
58 MISSING GROUP REPLICATOR
59 REPLICATOR NOT ALLOWED
60 MISSING RIGHT QUOTE
61 MISSING NUMBER AFTER APOSTROPHE
62 INVALID NUMERIC SPECIFICATION
63 UNBALANCED PARENTHESES
64 MISSING RIGHT PARENTHESIS

65 NUMBER > 255
66 MISSING D-PART
67 MISSING SEPARATOR
68 MISSING LEFT PARENTHESIS
69 INVALID C-SPECIFICATION
70 INVALID COMPLEX SPECIFICATION
71 REPLICATOR = ZERO
72 INVALID STRING SPECIFICATION
73 REPLICATOR > 255
74 STRING SPECIFICATION FOR NUMERIC ITEM
Attempt was made to output numeric item in PRINT USING with a string specification in format string.

75 NUMERIC SPECIFICATION FOR STRING ITEM
Attempt was made to output string item in PRINT USING with a numeric specification in format string.

86 I/O ERROR ON USER FILE

87 UNABLE TO OPEN LIST FILE

88 LIST FILE NOT TYPE ASCII

89 UNABLE TO OPEN INPUT FILE

90 INPUT FILE NOT TYPE ASCII

91 I/O ERROR ON LIST FILE

92 I/O ERROR ON INPUT FILE

93 I/O ERROR ON $STDLIST

94 NUMERIC FILE DATA FOR STRING ITEM
The destination variable in a READ# or MAT READ# statement was string but the next item in the file was numeric.

95 STRING FILE DATA FOR NUMERIC ITEM
The destination variable in a READ# or MAT READ# statement was numeric but the next item in the file was string.

96 STRING DATA FOR NUMERIC ITEM
The destination variable in a READ or MAT READ statement was numeric but the next item in the DATA list was string.

97 STRING UPDATE ITEM FOR NUMERIC DATA
The expression in the UPDATE# statement was string but the next item in the BASIC formatted file was numeric.

READER COMMENT SHEET

BASIC/3000 Compiler
Reference Manual
32103-90001 November 1974

We welcome your evaluation of this manual. Your comments and suggestions help us improve our publications. Please use additional pages if necessary.

Is this manual technically accurate?

Is this manual complete?

Is this manual easy to read and use?

Other comments?

FROM:

Name ______________________________

Company ______________________________

Address ______________________________

____

____

FOLD FOLD

FIRST CLASS
PERMIT NO. 1020
SANTA CLARA
CALIFORNIA

BUSINESS REPLY MAIL

No Postage Necessary if Mailed in the United States. Postage will be paid by

Publications Manager, Product Support Group
Hewlett-Packard Company
General Systems Division
5303 Stevens Creek Boulevard
Santa Clara, California 95050

FOLD FOLD

HEWLETT hp PACKARD

SALES & SERVICE OFFICES

AFRICA, ASIA, AUSTRALIA

ANGOLA
Telectra
Empresa Técnica de
Equipamentos
Eléctricos, S.A.R.L.
R. Barbosa Rodrigues, 42-I DT.
Caixa Postal, 6487
Luanda
Tel: 35515/6
Cable: TELECTRA Luanda

AUSTRALIA
Hewlett-Packard Australia
Pty. Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
P.O. Box 36
Doncaster East, Victoria 3109
Tel: 89-6351
Telex: 31-024
Cable: HEWPARD Melbourne

Hewlett-Packard Australia
Pty. Ltd.
31 Bridge Street
Pymble
New South Wales, 2073
Tel: 449-6566
Telex: 21561
Cable: HEWPARD Sydney

Hewlett-Packard Australia
Pty. Ltd.
153 Greenhill Road
Parkside, S.A., 5063
Tel: 272-5911
Telex: 82536 ADEL
Cable: HEWPARD ADELAID

Hewlett-Packard Australia
Pty.Ltd.
141 Stirling Highway
Nedlands, W.A. 6009
Tel: 86-5455
Telex: 93859 PERTH
Cable: HEWPARD PERTH

Hewlett-Packard Australia
Pty. Ltd.
121 Wollongong Street
Fyshwick, A.C.T. 2609
Tel: 95-2733
Telex: 62650 Canberra
Cable: HEWPARD CANBERRA

Hewlett Packard Australia
Pty. Ltd.
5th Floor
Teachers Union Building
495-499 Boundary Street
Spring Hill, 4000 Queensland
Tel: 229-1544
Cable: HEWPARD Brisbane

GUAM
Medical/Pocket Calculators Only
Guam Medical Supply, Inc.
Jay Ease Building, Room 210
P.O. Box 8947
Tamuning 96911
Tel: 646-4513
Cable: EARMED Guam

HONG KONG
Schmidt & Co.(Hong Kong) Ltd.
P.O. Box 297
Connalight Centre
39th Floor
Connaught Road, Central
Hong Kong
Tel: H-255291-5
Telex: 74766 SCHMC HX
Cable: SCHMIDTCO Hong Kong

INDIA
Blue Star Ltd.
Kasturi Buildings
Jamshedji Tata Rd
Bombay 400 020
Tel: 29 50 21
Telex: 001-2156
Cable BLUEFROST

Blue Star Ltd.
Sahas
414/2 Vir Savarkar Marg
Prabhadevi
Bombay 400 025
Tel: 45 78 87
Telex: 011-4093
Cable: FROSTBLUE

Blue Star Ltd.
Band Box House
Prabhadevi
Bombay 400 025
Tel: 45 73 01
Telex: 011-3751
Cable: BLUESTAR

Blue Star Ltd.
14/40 Civil Lines
Kanpur 208 001
Tel: 6 88 82
Telex: 292
Cable: BLUESTAR

Blue Star Ltd.
7 Hare Street
P.O. Box 506
Calcutta 700 001
Tel: 23-0131
Telex: 021-7655
Cable: BLUESTAR

Blue Star Ltd.
7th & 8th Floor
Bhandari House
91 Nehru Place
New Delhi 110024
Tel: 634770 & 635166
Telex: 031-2463
Cable: BLUESTAR

Blue Star Ltd.
Blue Star House
11/11A Magarath Road
Bangalore 560 025
Tel: 55668
Telex: 043-430
Cable: BLUESTAR

Blue Star Ltd.
Meeakshi Mandiran
xxx/1678 Mahatma Gandhi Rd.
Cochin 682 016
Tel: 32069,32161,32282
Telex: 0885-514
Cable: BLUESTAR

Blue Star Ltd.
1-1-117/1
Sarojini Devi Road
Secunderabad 500 003
Tel: 70126, 70127
Cable: BLUEFROST
Telex: 015-459

Blue Star Ltd.
2/34 Kodambakkam High Road
Madras 600034
Tel: 82056
Telex: 041-379
Cable: BLUESTAR

Blue Star Ltd.
Nataraj Mansions
2nd Floor Bistupur
Jamshedpur 831 001
Tel: 7383
Cable: BLUESTAR
Telex. 240

INDONESIA
BERCA Indonesia P.T.
P.O. Box 496/Jkt.
JLN●Abdul Muis 62
Jakarta
Tel: 40369, 49886,49255,356038
JKT 42895
Cable: BERCACON

BERCA Indonesia P.t.
63 JL. Raya Gubeng
Surabaya
Tel: 44309

ISRAEL
Electronics & Engineering Div.
of Motorola Israel Ltd.
17, Kremenetski Street
P.O. Box 25016
Tel-Aviv
Tel: 38973
Telex: 33569
Cable: BASTEL Tel-Aviv

JAPAN
Yokogawa-Hewlett-Packard Ltd.
Ohashi Building
59-1 Yoyogi 1-Chome
Shibuya-ku, **Tokyo** 151
Tel: 03-370-2281/92
Telex: 232-2024YHP
Cable: YHPMARKET TOK 23-724

Yokogawa-Hewlett-Packard Ltd.
Chuo Bldg., 4th Floor
4-20, Nishinakajima 5-chome
Yodogawa-ku, Osaka-shi
Osaka,532
Tel: 06-304-6021

Yokogawa-Hewlett-Packard Ltd.
Nakamo Building
24 Kami Sasajima-cho
Nakamura-ku, **Nagoya** , 450
Tel: (052) 571-5171

Yokogawa-Hewlett-Packard Ltd.
Tanigawa Building
2-24-1 Tsuruya-cho
Kanagawa-ku
Yokohama, 221
Tel: 045-312-1252
Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.
Mito Mitsu Building
105, Chome-1,San-no-maru
Mito, Ibaragi 310
Tel: 0292-25-7470

Yokogawa-Hewlett-Packard Ltd.
Inoue Building
1348-3, Asahi-cho, 1-chome
Atsugi, Kanagawa 243
Tel: 0462-24-0452

Yokogawa-Hewlett-Packard Ltd.
Kumagaya Asahi
Hachijuni Building
4th Floor
3-4, Tsukuba
Kumagaya, Saitama 360
Tel: 0485-24-6563

KENYA
Technical Engineering
Services(E.A.)Ltd.,
P.O. Box 18311
Nairobi
Tel: 557726/556762
Cable: PROTON

Medical Only
International Aeradio(E.A.)Ltd.,
P.O. Box 19012
Nairobi Airport
Nairobi
Tel: 336055/56
Telex: 22201/22301
Cable: INTAERIO Nairobi

KOREA
Samsung Electronics Co., Ltd.
20th Fl. Dongbang Bldg. 250, 2-KA
C.P.O. Box 2775
Taepyung-Ro, Chung-Ku
Seoul
Tel: (23) 6811
Telex: 22575
Cable: ELEKSTAR Seoul

MALAYSIA
Teknik Mutu Sdn. Bhd.
2 Lorong 13/6A
Section 13
Petaling Jaya,**Selangor**
Tel: 54994/54916
Telex: MA 37605

Protel Engineering
P.O. Box 1917
Lot 259, Satok Road
Kuching, **Sarawak**
Tel: 2400
Cable: PROTEL ENG

MOZAMBIQUE
A.N. Goncalves, Lta.
162, 1 Apt. 14 Av. D. Luis
Caixa Postal 107
Lourenco Marques
Tel: 27091, 27114
Telex: 6-203 NEGON Mo
Cable: NEGON

NEW ZEALAND
Hewlett-Packard (N.Z.) Ltd.
P.O. Box 9443
Courtenay Place
Wellington
Tel: 877-199
Telex: NZ 3839
Cable: HEWPACK Wellington

Hewlett-Packard (N.Z.) Ltd.
Pakuranga Professional Centre
267 Pakuranga Highway
Box 51092
Pakuranga
Tel: 569-651
Cable: HEWPACK,Auckland

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
Scientific Division
79 Carlton Gore Rd., Newmarket
P.O. Box 1234
Auckland
Tel: 75-289
Cable:DENTAL Auckland

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
P.O. Box 1994
147-161 Tory St.
Wellington
Tel: 850-799
Telex: 3858
Cable: DENTAL, Wellington

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
P.O. Box 309
239 Stanmore Road
Christchurch
Tel: 892-019
Cable: DENTAL, Christchurch

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
303 Great King Street
P.O. Box 233
Dunedin
Tel: 88-817
Cable: DENTAL, Dunedin

NIGERIA
The Electronics
Instrumentations Ltd.
N6B/770 Oyo Road
Oluseun House
P.M.B. 5402
Ibadan
Tel: 61577
Telex: 31231 TEIL Nigeria
Cable: THETEIL Ibadan

The Electronics Instrumentations Ltd.
144 Agege Motor Road, Mushin
P.O. Box 6645
Lagos
Cable: THETEIL Lagos

PAKISTAN
Mushko & Company, Ltd.
Oosman Chambers
Abdullah Haroon Road
Karachi-3
Tel: 511027, 512927
Telex: 2894
Cable: COOPERATOR Karachi

Mushko & Company, Ltd.
38B, Satellite Town
Rawalpindi
Tel: 41924
Cable: FEMUS Rawalpindi

PHILIPPINES
The Online Advanced
Systems Corporation
8th Floor, Filcapital Bldg.
Ayala Avenue
Makati, Metro**Manila**
Tel: 85-35-81, 85-34-91
Telex: 3274 ONLINE

RHODESIA
Field Technical Sales
45 Kelvin Road North
P.O. Box 3458
Salisbury
Tel: 705231 (5 lines)
Telex: RH 4122

SINGAPORE
Hewlett-Packard Singapore
(Pte.) Ltd.
1150 Depot Road
Alexandra P.O. Box 58
Singapore 4
Tel: 270-2355
Telex: HPSG RS 21486
Cable: HEWPACK, Singapore

SOUTH AFRICA
Hewlett-Packard South Africa
(Pty.). Ltd.
Private Bag Wendywood
Sandton, Transvaal 2144
Hewlett-Packard Centre
Daphne Street, Wendywood,
Sandton, Transvaal 2144
Tel: 802-10408
Telex: 8-4782
Cable: HEWPACK JOHANNESBURG

Service Department
Hewlett-Packard South Africa
(Pty.). Ltd.
P.O. Box 39325
Gramley, Sandton, 2018
451 Wynberg Extension 3,
Sandton, 2001
Tel: 636-8188/9
Telex: 8-2391

Hewlett-Packard South Africa
(Pty.). Ltd.
P.O. Box 120
Howard Place, Cape Province, 7450
Pine Park Centre, Forest Drive,
Pinelands, Cape Province, 7405
Tel: 53-7955 thu 9
Telex: 57-0006

Service Department
Hewlett-Packard South Africa
(Pty.). Ltd.
P.O. Box 37099
Overport, Durban 4067
Braby House
641 Ridge Road
Durban, 4001
Tel: 88-7478
Telex: 6-7954

TAIWAN
Hewlett-Packard Far East Ltd.,
Taiwan Branch
39 Chung Hsiao West Road
Sec. 1, 7th Floor
Taipei
Tel: 3819160-4
Cable: HEWPACK TAIPEI

Hewlett-Packard Far East Ltd.
Taiwan Branch
68-2, Chung Cheng 3rd. Road
Kaohsiung
Tel: (07) 242318-Kaohsiung

Analytical Only
San Kwang Instruments Co., Ltd.,
No. 20, Yung Sui Road
Taipei
Tel: 3715171-4 (5 lines)
Telex: 22894 SANKWANG
Cable: SANKWANG TAIPEI

TANZANIA
Medical Only
International Aeradio (E.A.), Ltd.
P.O. Box 861
Dar es Salaam
Tel: 21251 Ext. 265
Telex: 41030

THAILAND
UNIMESA Co., Ltd.
Elcom Research Building
2538 Sukumvit Ave.
Bangkok
Tel: 3932387, 3930338
Cable: UNIMESA Bangkok

UGANDA
Medical Only
International Aeradio(E.A.), Ltd.,
P.O. Box 2577
Kampala
Tel: 54388
Cable: INTAERIO Kampala

ZAMBIA
R.J. Tilbury (Zambia) Ltd.
P.O. Box 2792
Lusaka
Tel: 73793
Cable: ARJAYTEE, Lusaka

OTHER AREAS NOT LISTED, CONTACT:
Hewlett-Packard Intercontinental
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 493-1501
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

CANADA

ALBERTA
Hewlett-Packard (Canada) Ltd.
11620A - 168th Street
EdmontonT5M 3T9
Tel: (403) 452-3670
TWX: 610-831-2431 EDTH

Hewlett-Packard (Canada) Ltd.
210,7220 Fisher St. S.E.
Calgary T2H 2H8
Tel: (403) 253-2713
Twx: 610-821-6141

BRITISH COLUMBIA
Hewlett-Packard (Canada) Ltd.
837 E. Cordova Street
Vancouver V6A 3R2
Tel: (604) 254-0531
TWX: 610-922-5059 VCR

MANITOBA
Hewlett-Packard (Canada) Ltd.
513 Century St.
St. James
Winnipeg R3H 0L8
Tel: (204) 786-7581
TWX: 610-671-3531

NOVA SCOTIA
Hewlett-Packard (Canada) Ltd.
800 Windmill Road
Dartmouth B3B 1L1
Tel: (902) 469-7820
TWX: 610-271-4482 HFX

ONTARIO
Hewlett-Packard (Canada) Ltd.
1020 Morrison Dr.
Ottawa K2H 8K7
Tel: (613) 820-6483
TWX: 610-563-1636

Hewlett-Packard (Canada) Ltd.
6877 Goreway Drive
Mississauga L4V 1M8
Tel: (416) 678-9430
TWX: 610-492-4246

QUEBEC
Hewlett-Packard (Canada) Ltd.
275 Hymus Blvd.
Pointe Claire H9R 1G7
Tel: (514) 697-4232
TWX: 610-422-3022
TLX: 05-821521 HPCL

FOR CANADIAN AREAS NOT LISTED:
Contact Hewlett-Packard (Canada)
Ltd. in Mississauga

CENTRAL AND SOUTH AMERICA

ARGENTINA
Hewlett-Packard Argentina
S.A.
Av. Leandro N. Alem 822 - 12
1001**Buenos Aires**
Tel: 31-6063,4,5,6 and 7
Telex: Public Booth N 9
Cable: HEWPACK ARG

BOLIVIA
Casa Kavlin S.A.
Calle Potosi 1130
P.O. Box 500
La Paz
Tel: 41530,53221
Telex: CWC BX 5298,ITT 3560082
Cable: KAVLIN

BRAZIL
Hewlett-Packard do Brasil
I.e.C. Ltda.
Avenida Rio Negro, 980
Alphaville
06400**Barueri** SP
Tel: 429-2148/9;429-2118/9

Hewlett-Packard do Brasil
I.e.C. Ltda.
Rua Padre Chagas, 32
90000-**Pôrto Alegre**-RS
Tel: (0512) 22-2998, 22-5621
Cable: HEWPACK Pôrto Alegre

Hewlett-Packard do Brasil
I.E.C. Ltda.
Rua Siqueira Campos, 53
Copacabana
20000-**Rio de Janeiro**
Tel: 257-80-94-DDD (021)
Telex: 391-212-1905 HEWP-BR
Cable: HEWPACK
Rio de Janeiro

CHILE
Calcagni y Metcalfe Ltda.
Alameda 580-Of. 807
Casilla 2118
Santiago, 1
Tel: 398613
Telex: 3520001 CALMET
Cable: CALMET Santiago

COLOMBIA
Instrumentación
Henrik A. Langebaek & Kier S.A.
Carrera 7 No. 48-75
Apartado Aéreo 6287
Bogotá, I.D.E.
Tel: 69-88-77
Cable: AARIS Bogotá
Telex: 044-400

COSTA RICA
Científica Costarricense S.A.
Calle Central, Avenidas 1 y 3
Apartado 10159
San José
Tel: 21-86-13
Cable: GALGUR San José

ECUADOR
Medical Only
A.F. Vizcaíno Compañía Ltda.
Av. Rio Amazonas No. 239
P.O. Box 2925
Quito
Tel: 242-150,247-033/034
Cable: Astor Quito

Calculators Only
Computadoras y Equipos
Electrónicos
P.O. Box 6423 CCI
Eloy Alfaro #1824,3 Piso
Quito
Tel: 453482
Telex: 02-2113 Sagita Ed
Cable: Sagita-Quito

EL SALVADOR
Instrumentacion y Procesamiento
Electronico de el Salvador
Bulevar de los Heroes II-48
San Salvador
Tel: 252787

GUATEMALA
IPESA
Avenida La Reforma 3-48,
Zona 9
Guatemala City
Tel: 63627, 64786
Telex: 4192 Teletro Gu

MEXICO
Hewlett-Packard Mexicana,
S.A. de C.V.
Torres Adalid No. 21, 11 Piso
Col. del Valle
Mexico 12, D.F.
Tel: (905) 543-42-32
Telex: 017-74-507

Hewlett-Packard Mexicana,
S.A. de C.V.
Ave. Constitución No. 2184
Monterrey, N.L.
Tel: 48-71-32, 48-71-84
Telex: 038-843

NICARAGUA
Roberto Terán G.
Apartado Postal 689
Edificio Terán
Managua
Tel: 25114, 23412,23454
Cable: ROTERAN Managua
Calculators Only
Cientifice Costarricewse S.A.
Ciudad Jardin D-1
Managua
Tel: 24108

PANAMA
Electrónico Balboa, S.A.
P.O. Box 4929
Calle Samuel Lewis
Cuidad de Panama
Tel: 64-2700
Telex: 3431103 Curunda,
Canal Zone
Cable: ELECTRON Panama

PARAGUAY
Z.J. Melamed S.R.L.
División: Aparatos y Equipos
Médicos
División: Aparatos y Equipos
Científicos y de Investigación
P.O.Box 676
Chile-482, Edificio Victoria
Asunción
Tel: 91-271, 91-272
Cable: RAMEL

PERU
Compañia Electro Médica S.A.
Los Flamencos 145
San Isidro Casilla 1030
Lima 1
Tel: 41-4325
Cable: ELMED Lima

PUERTO RICO
Hewlett-Packard Inter-Americas
Puerto Rico Branch Office
Calle 272,
No. 203 Urb. Country Club
Carolina 00924
Tel: (809) 762-7255
Telex: 345 0514

URUGUAY
Pablo Ferrando S.A.
Comercial e Industrial
Avenida Italia 2877
Casilla de Correo 370
Montevideo
Tel: 40-3102
Cable: RADIUM Montevideo

VENEZUELA
Hewlett-Packard de Venezuela
C.A.
P.O. Box 50933
Caracas 105
Los Ruices Norte
3a Transversal
Edificio Segre
Caracas 107
Tel: 35-00-11 (20 lines)
Telex: 25146 HEWPACK
Cable: HEWPACK Caracas

FOR AREAS NOT LISTED, CONTACT:
Hewlett-Packard
Inter-Americas
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 493-1501
TWX: 910-373-1260
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

EUROPE, NORTH AFRICA AND MIDDLE EAST

AUSTRIA
Hewlett-Packard Ges.m.b.H
Handelskai 52
P.O. box 7
A-1205 **Vienna**
Tel: (0222) 351621 to 27
cable: HEWPAK Vienna
Telex: 75923 hewpak a

BELGIUM
Hewlett-Packard Benelux
S.A./N.V.
Avenue de Col-Vert, 1,
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 672 22 40
Cable: PALOBEN Brussels
Telex: 23 494 paloben bru

CYPRUS
Kypronics
19, Gregorios & Xenopoulos Rd
P.O. Box 1152
CY-**Nicosia**
Tel: 45628/29
Cable: KYPRONICS PANDEHIS
Telex: 3018

CZECHOSLOVAKIA
Vyvojova a Provozni Zakladna
Vyzkumnych Ustavu v Bechovicich
CSSR-25097 **Bechovice u Prahy**
Tel: 89 93 41
Telex: 121333

Institute of Medical Bionics
Vyskumny Ustav Lekarskej Bioniky
Jedlova 6
CS-88346 **Bratislava-Kramare**
Tel. 44-551/45-541

DDR
Entwicklungslabor der TU Dresden
Forschungsinstitut Meinsberg
DDR-7305
Waldheim/Meinsberg
Tel: 37 667
Telex: 112145

Export Contact AG Zuerich
Guenther Forgber
Schlegelstrasse 15
1040 **Berlin**
Tel: 42-74-12
Telex: 111889

DENMARK
Hewlett-packard A/S
Datavej 52
DK-3460 **Birkerød**
Tel: (02) 81 66 40
Cable: HEWPACK AS
Telex: 166 40 hpas

Hewlett-Packard A/S
Navervej 1
DK-8600 **Silkeborg**
Tel: (06) 82 71 66
Telex: 166 40 hpas
Cable: HEWPACK AS

FINLAND
Hewlett-Packard OY
Nahkahousuntie 5
P.O. Box 6
SF-00211 **Helsinki** 21
Tel: (90) 6923031
Cable: HEWPACKOY Helsinki
Telex: 12-1563 HEWPA SF

FRANCE
Hewlett-Packard France
Quartier de Courtaboeuf
Boite Postale No. 6
F-91401 **Orsay** Cédex
Tel: (1) 907 78 25
Cable: HEWPACK Orsay
Telex: 600048

Hewlett-Packard France
Agency Régionale
"Le Saquin"
Chemin des Mouilles
B.P. 162
F-69130 **Ecully**
Tel: (78) 33 81 25,
Cable: HEWPACK Eculy
Telex: 31 06 17

Hewlett-Packard France
Agence Régionale
Péricentre de la Cépière
Chemin de la Cépière, 20
F-31300 **Toulouse-Le Mirail**
Tel:(61) 40 11 12
Cable: HEWPACK 51957
Telex: 510957

Hewlett-Packard France
Agence Régionale
Aéroport principal de
Marseille-Marignane
F-13721**Marignane**
Tel: (91) 89 12 36
Cable: HEWPACK MARGN
Telex: 410770

Hewlett-Packard France
Agence Régionale
63, Avenue de Rochester
B.P. 1124
F-35014 **Rennes** Cédex
Tel: (99) 36 33 21
Cable: HEWPACK 74912
Telex: 740912

Hewlett-Packard France
Agence Régionale
74, Allée de la Robertsau
F-67000 **Strasbourg**
Tel: (88) 35 23 20/21
Telex: 890141
Cable: HEWPACK STRBG

Hewlett-Packard France
Agence Régionale
Centre Vauban
201, rue Colbert
Entrée A2
F-59000 **Lille**
Tel: (20) 51 44 14
Telex: 820744

Hewlett-Packard France
Centre d' Affaires Paris-Nord
Bâtiment Ampère
Rue de La Commune de Paris
B.P. 300
F-93153 **Le Blanc Mesnil Cédex**
Tel: (01) 931 88 50

GERMAN FEDERAL REPUBLIC
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Bernerstrasse 117
Postfach 560 140
D-6000 **Frankfurt** 56
Tel: (0611) 50 04-1
Cable: HEWPACKSA Frankfurt
Tel: (0611) 50 04-1
Cable: HEWPACKSA Frankfurt
Telex: 04 13249 hpffmd

Hewlett-Packard GmbH
Technisches Buero Böblingen
Herrenbergerstrasse 110
D-7030 **Böblingen**, Württemberg
Tel: (07031) 667-1
Cable: HEPAK Böblingen
Telex: 07265739 bbn

Hewlett-Packard GmbH
Technisches Buero Düsseldorf
Emanuel-Leutze-Str 1 (Seestern)
D-4000 **Düsseldorf** 11
Tel: (0211) 59711
Telex: 085/86 533 hpdd d

Hewlett-Packard GmbH
Technisches Buero Hamburg
Wendenstrasse 23
D-2000 **Hamburg** 1
Tel: (040) 24 13 93
Cable: HEWPACKSA Hamburg
Telex: 21 63 032 hphh d

Hewlett-Packard GmbH
Technisches Buero Hannover
Am Grossmarkt 6
D-3000 **Hannover-Kleefeld 91**
Tel: (0511) 46 60 01
Telex: 092 3259

Hewlett-Packard GmbH
Werk Groetzingen
Ohmstrasse 6
D-7500 **Karlsruke 41**
Tel: (0721) 69 40 06
Telex: 07-825707

Hewlett-Packard GmbH
Technisches Buero Nuremberg
Neumeyer Str. 90
D-8500 **Nuremberg**
Tel: (0911) 56 30 83/85
Telex: 0623 860

Hewlett-Packard GmbH
Technisches Buero München
Unterhachinger Strasse 28
ISAR Center
D-8012 **Ottobrunn**
Tel: (089) 601 30 61/7
Cable: HEWPACKSA München
Telex: 0524985

Hewlett-Packard GmbH
Technisches Buero Berlin
Keith Strasse 2-4
D-1000 **Berlin** 30
Tel: (030) 24 90 86
Telex: 18 3405 hpbln d

GREECE
Kostas Karayannis
18, Ermou Street
GR-**Athens** 126
Tel: 3237731
Cable: RAKAR Athens
Telex: 21 59 62 rkar gr

Analytical Only
"INTECO" G. Papathanassiou & Co.
Marni 17
GR - **Athens** 103
Tel: 522 1915
Cable: INTEKNIKA Athens
Telex: 21 5329 INTE GR

Medical Only
Technomed Hellas Ltd.
52,Skoufa Street
GR - **Athens** 135
Tel: 362 6972, 363 3830
Cable:etalak athens
Telex: 21-4693 ETAL GR

HUNGARY
MTA
Müszerügyi és Méréstechnikai
Szolgalata
Lenin Krt. 67
Tel. 42 03 38
Telex: 22 51 14

ICELAND
Medical Only
Elding Trading Company Inc.
Hafnarhvoli - Tryggvatotu
IS-**Reykjavik**
Tel: 1 58 20
Cable: ELDING Reykjavik

IRAN
Hewlett-Packard Iran Ltd.
No. 13, Fourteenth St.
Miremad Avenue
P.O. Box 41/2419
IR-**Tehran**
Tel: 851082-7
Telex: 213405 HEWP IR

IRAQ
Hewlett-Packard Trading Co.
Mansoor City
Baghdad
Tel: 5517827
Telex: 2455 Hepairaq ik
Cable: HEWPACDAD,
Baghdad Iraq

IRELAND
Hewlett-Packard Ltd.
King Street Lane
GB-**Winnersh,**Wokingham
Berks, RG11 5AR
Tel: (0734) 78 47 74
Telex: 847178/848179

ITALY
Hewlett-Packard Italiana S.p.A.
Via Amerigo Vespucci 2
Casella postale 3645
I-20100 **Milano**
Tel: (2) 6251 (10 lines)
Cable: HEWPACKIT Milano
Telex: 32046

Hewlett-Packard Italiana S.p.A.
Via Pietro Maroncelli 40
(ang. Via Visentin)
I-35100 **Padova**
Tel: (49) 66 48 88
Telex: 41612 Hewpacki

Medical only
Hewlett-Packard Italiana S.p.A.
Via d'Aghiardi, 7
I-56100 **Pisa**
Tel: (050) 2 32 04
Telex: 32046 via Milano

Hewlett-Packard Italiana S.p.A.
Via G. Armellini 10
I-00143 **Roma**
Tel: (06) 54 69 6l
Telex: 61514
Cable: HEWPACKIT Roma

Hewlett-Packard Italiana S.p.A.
Corso Giovanni Lanza 94
I-10130 **Torino**
Tel: (011) 682245/659308

Medical/Calculators Only
Hewlett-Packard Italiana S.p.A.
Via Principe Nicola 43 G/C
I-95126 **Catania**
Tel:(095) 37 05 04

Hewlett-Packard Italiana S.p.A.
Via Amerigo Vespucci, 9
I-80142 **Napoli**
Tel: (081) 33 77 11

Hewlett-Packard Italiana S.p.A.
Via E. Masi, 9/B
I-40137 **Bologna**
Tel: (051) 30 78 87

KUWAIT
Al-Khaldiya Trading &
Contracting Co.
P.O. Box 830
Kuwait
Tel: 42 49 10
Cable: VISCOUNT

LUXEMBURG
Hewlett-Packard Benelux
S.A./N.V.
Avenue du Col-Vert, 1,
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 672 22 40
Cable: PALOBEN Brussels
Telex: 23 494

MOROCCO
Gerep
190, Blvd. Brahim Roudani
Casablanca
Tel: 25-16-76/25-90-99
Cable: Gerep-Casa
Telex: 23739

NETHERLANDS
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 121
P.O. Box 667
NL-1134 **Amstelveen**
Tel: (020) 47 20 21
Cable: PALOBEN Amsterdam
Telex: 13 216 hepa nl

NORWAY
Hewlett-Packard Norge A/S
Nesveien 13
Box 149
N-1344 **Haslum**
Tel: (02) 53 83 60
Telex: 16621 hpnas n

POLAND
Biuro Informacji Technicznej
Hewlett-Packard
U1 Stawki 2, 6P
00-950**Warszawa**
Tel: 395962/395187
Telex: 81 24 53 hepa pl

UNIPAN
Zaklad Doswiadczalny
Budowy Aparatury Naukowej
U1. Krajowej Rady Narodowej 51/55
00-800 **Warszawa**
Tel: 36190
Telex: 81 46 48

Zaklady Naprawcze Sprzetu
Medycznego
Plac Komuny Paryskiej 6
90-007 **Lódź**
Tel: 334-41, 337-83

PORTUGAL
Telectra-Empresa Técnica de
Equipamentos Eléctricos S.a.r.l
Rua Rodrigo da Fonseca 103
P.O. Box 2531
P-**Lisbon** 1
Tel: (19) 68 60 72
Cable: TELECTRA Lisbon
Telex: 12598

Medical only
Mundinter
Intercambio Mundial de Comércio
S.a.r.l.
Av.A.A.de Aguiar 138
P.O. Box 2761
P - **Lisbon**
Tel: (19) 53 21 31/7
Cable: INTERCAMBIO Lisbon

RUMANIA
Hewlett-Packard Reprezentanta
Bd.N. Balcescu 16
Bucharest
Tel: 158023/138885
Telex: 10440

I.I.R.U.C.
Intreprinderea Pentru
Intretinerea
Si Repararea Utilajelor de Calcul
B-dul prof. Dimitrie Pompei 6
Bucharest-Sectorul 2
Tel: 12 64 30
Telex: 11716

SAUDI ARABIA
Modern Electronic Establishment
King Abdul Aziz str.(Head office)
P.O. Box 1228
Jeddah
Tel: 31173-332201
Cable: ELECTRA

P.O. Box 2728 (Service center)
Riyadh
Tel: 62596-66232
Cable: RAOUFCO

SPAIN
Hewlett-Packard Española, S.A.
Jerez, Calle 3
E-**Madrid** 16
Tel:(1) 458 26 00 (10 lines)
Telex: 23515 hpe

Hewlett-Packard Española, S.A.
Milanesado 21-23
E-**Barcelona** 17
Tel: (3) 203 6200 (5 lines)
Telex: 52603 hpbe e

Hewlett-Packard Española, S.A.
Av Ramón y Cajal, 1
Edificio Sevilla, planta 9,
E-**Seville** 5
Tel: 64 44 54/58

Hewlett-Packard Española S.A.
Edificio Albia II 7 B
E-**Bilbao**-1
Tel: 23 83 06/23 82 06

Calculators Only
Hewlett-Packard Española S.A.
Gran Via Fernando El Católico, 67
E-**Valencia**-8
Tel: 326 67 28/326 85 55

SWEDEN
Hewlett-Packard Sverige AB
Enighetsvägen 1-3
Fack
S-161 20 **Bromma** 20
Tel: (08) 730 05 50
Cable: MEASUREMENTS
Stockholm
Telex: 10721

Hewlett-Packard Sverige AB
Ostra Vintergatan 22
S-702 40 **Orebro**
Tel: (019) 14 07 20

Hewlett-Packard Sverige AB
Frötallsgatan 30
S-421 32 **Västra Frölunda**
Tel: (031) 49 09 50
Telex: 10721 Via Bromma Office

SWITZERLAND
Hewlett-Packard (Schweiz) AG
Zürcherstrasse 20
P.O. Box 307
CH-8952 **Schlieren-Zurich**
Tel: (01) 730 52 40/730 18 21
Cable: HPAG CH
Telex: 53933 hpag ch

Hewlett-Packard (schweiz) AG
Château Bloc 19
CH-1219 **Le Lignon-Geneva**
Tel: (022) 96 03 22
Cable: HEWPACKAG Geneva
Telex: 27 333 hpag ch

SYRIA
Medical/Calculator only
Sawah & Co.
Place Azmé
B.P. 2308
SYR-**Damascus**
Tel: 16367, 19697, 14268
Cable: SAWAH, Damascus

TURKEY
Telekom Engineering Bureau
P.O. Box 437
Beyoğlu
TR-**Istanbul**
Tel: 49 40 40
Cable: TELEMATION Istanbul
Telex: 23609

Medical only
E.M.A.
Muhendislik Kollektif Sirketi
Adakale Sokak 41/6
TR-**Ankara**
Tel: 175622

Analytical only
Yilmaz Ozyurek
Milli Mudafaa Cad No. 16/6
Kizilay
TR-**Ankara**
Tel: 25 03 09
Telex: 42576 Ozek tr

UNITED KINGDOM
Hewlett-Packard Ltd.
King Street Lane
GB-**Winnersh,** Wokingham
Berks. RG11 5AR
Tel: (0734) 78 47 74
Cable: Hewpie London
Telex:847178/9

Hewlett-Packard Ltd.
Trafalger House,
Navagation Road
Altrincham
Cheshire WA14 1NU
Tel: (061) 928 6422
Cable: Hewpie Manchester
Telex: 668068

Hewlett-Packard Ltd.
Lygon Court
Hereward Rise
Dudley Road
Halesowen,
West Midlands B62 8SD
Tel: (021) 550 9911
Telex: 339105

Hewlett-Packard Ltd.
Wedge House
799, London Road
GB-**Thornton Heath**
Surrey CR4 6XL
Tel: (01) 684 0103/8
Telex: 946825

Hewlett-Packard Ltd.
c/o Makro
South Serviceholesale Centre
Wear Industrial Estate
Washington
GB-**New Town**, County Durham
Tel: Washington 464001 ext. 57/58

Hewlett-Packard Ltd
10, Wesley St.
GB-**Castleford**
West Yorkshire WF10 1AE
Tel: (09775) 50402
Telex: 557355

Hewlett-Packard Ltd
1, Wallace Way
GB-**Hitchin**
Herts
Tel: (0462) 52824/56704
Telex: 825981

USSR
Hewlett-Packard
Representative Office USSR
Pokrovsky Boulevard 4/17-KW 12
Moscow 101000
Tel:294-2024
Telex: 7825 hewpak su

YUGOSLAVIA
Iskra-standard/Hewlett-Packard
Miklosiceva 38/VII
61000 **Ljubljana**
Tel: 31 58 79/32 16 74
Telex: 31583

SOCIALIST COUNTRIES NOT SHOWN PLEASE CONTACT:
Hewlett-Packard Ges.m.b.H
P.O. Box 7
A-1205 **Vienna**, Austria
Tel: (0222) 35 16 21 to 27
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

MEDITERRANEAN AND MIDDLE EAST COUNTRIES NOT SHOWN PLEASE CONTACT:
Hewlett-Packard S.A.
Mediterranean and Middle
East Operations
35, Kolokotroni Street
Platia Kefallariou
GR-Kifissia-**Athens**, Greece
Tel: 8080337/359/429
Telex: 21-6588
Cable: HEWPACKSA Athens

FOR OTHER AREAS NOT LISTED CONTACT
Hewlett-Packard S.A.
7, rue du Bois-du-Lan
P.O. Box
CH-1217 Meyrin 2 - **Geneva**
Switzerland
Tel: (022) 82 70 00
Cable: HEWPACKSA Geneva
Telex: 2 24 86

UNITED STATES

ALABAMA
8290 Whitesburg Dr., S.E.
P.O. Box 4207
Huntsville 35802
Tel: (205) 881-4591

Medical Only
228 W. Valley Ave.,
Room 220
Birmingham 35209
Tel: (205) 942-2081/2

ARIZONA
2336 E. Magnolia St.
Phoenix 85034
Tel: (602) 244-1361

2424 East Aragon Rd.
Tucson 85706
Tel: (602) 294-3148

·ARKANSAS
Medical Service Only
P.O. Box 5646
Brady Station
Little Rock 72215
Tel: (501) 376-1844

CALIFORNIA
1430 East Orangethorpe Ave.
Fullerton 92631
Tel: (714) 870-1000

3939 Lankershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2671

6305 Arizona Place
Los Angeles 90045
Tel: (213) 649-2511
TWX: 910-328-6147

·Los Angeles

·Ridgecrest
Tel: (714) 446-6165

646 W. North Market Blvd.
Sacramento 95834
Tel: (916) 929-7222

9606 Aero Drive
P.O. Box 23333
San Diego 92123
Tel: (714) 279-3200

COLORADO
5600 South Ulster Parkway
Englewood 80110
Tel: (303) 771-3455

CONNECTICUT
12 Lunar Drive
New Haven 06525
Tel: (203) 389-6551
TWX: 710-465-2029

FLORIDA
P.O. Box 24210
2806 W. Oakland Park Blvd.
Ft. Lauderdale 33311
Tel: (305) 731-2020

·Jacksonville
Medical Service only
Tel: (904) 398-0663

P.O. Box 13910
6177 Lake Ellenor Dr.
Orlando 32809
Tel: (305) 859-2900

P.O. Box 12826
Pensacola 32575
Tel: (904) 476-8422

GEORGIA
P.O. Box 105005
Atlanta 30348

P.O. Box 2103
Warner Robins 31098
Tel: (912) 922-0449

HAWAII
2875 So. King Street
Honolulu 96814
Tel: (808) 955-4455
Telex: 723-705

ILLINOIS
5201 Tollview Dr.
Rolling meadows 60008
Tel: (312) 255-9800
TWX: 910-687-2260

INDIANA
7301 North Shadeland Ave.
Indianapolis46250
Tel: (317)842-1000
TWX: 810-260-1797

IOWA
1902 Broadway
Iowa City 52240
Tel: (319) 338-9466

KENTUCKY
Medical Only
Atkinson Square
3901 Atkinson Dr.,
Suite 207
Louisville 40218
Tel: (502) 456-1573

LOUISIANA
P.O. Box 840
3229-39 Williams Boulevard
Kenner 70063
Tel: (504) 443-6201

MARYLAND
6707 Whitestone Road
Baltimore 21207
Tel: (301) 944-5400

MASSACHUSETTS
32 Hartwell Ave.
Lexington 02173
Tel: (617) 861-8960
TWX: 710-326-6904

MICHIGAN
23855 Research Drive
Farmington Hills 48024
Tel: (313) 476-6400

MINNESOTA
2400 N. Prior Ave.
St. Paul 55113
Tel: (612) 636-0700

MISSISSIPPI
·Jackson
Medical Service only
Tel: (601) 982-9363

MISSOURI
11131 Colorado Ave.
Kansas City 64137
Tel: (816) 763-8000
TWX: 910-771-2087

1024 Executive Parkway
St. Louis 63141
Tel: (314) 878-0200

NEBRASKA
Medical Only
7171 Mercy Road
Suite 110
Omaha 68106
Tel: (402) 392-0948

NEW JERSEY
W. 120 Century Rd.
Paramus 07652
Tel: (201) 265-5000

NEW MEXICO
P.O. Box 11634
Station E
11300 Lomas Blvd., N.E.
Albuquerque 87123
Tel: (505) 292-1330
TWX: 910-989-1185

156 Wyatt Drive
Las Cruces 88001
Tel: (505) 526-2484
TWX: 910-9983-0550

NEW YORK
6 Automation Lane
Computer Park
Albany 12205
Tel: (518) 458-1550

201 South Avenue
Poughkeepsie 12601
Tel: (914) 454-7330
TWX: 510-253-5981

650 Perinton Hill Office Park
Fairport 14450
Tel: (716) 223-9950

5858 East Molloy Road
Syracuse 13211
Tel: (315) 454-2486
TWX: 710-541-0482

1 Crossways Park West
Woodbury 11797
Tel: (516) 921-0300
TWX: 710-990-4951

NORTH CAROLINA
P.O. Box 5188
1923 North Main Street
High Point 27262
Tel: (919) 885-8101

OHIO
16500 Sprague Road
Cleveland 44130

1041 Kingsmill Parkway
Columbus 43229
Tel: (614) 436-1041

OKLAHOMA
P.O. Box 32008
Oklahoma City 73132
Tel: (405) 721-0200

OREGON
17890 SW Lower Boones
Ferry Road
Tualatin 97062
Tel: (503) 620-3350

PENNSYLVANIA
111 Zeta Drive
Pittsburgh 15238
Tel: (412) 782-0400

1021 8th Avenue
King of Prussia Industrial Park
King of Prussia 19406
Tel: (215) 265-7000
TWX: 510-660-2670

SOUTH CAROLINA
6941-0 N. Trenholm Road
Columbia 29260
Tel: (803) 782-6493

TENNESSEE
·Knoxville
Medical Service only
Tel: (615) 523-5022

3027 Vanguard Dr.
Director's Plaza
Memphis 38131
Tel: (901) 346-8370

Nashville
Medical Service only
Tel: (615) 244-5448

10535 Harwin Dr.
Houston 77036
Tel: (713) 776-6400

205 Billy Mitchell Road
San Antonio 78226
Tel: (512) 434-8241

UTAH
2160 South 3270 West Street
Salt Lake City 84119
Tel: (801) 972-4711

VIRGINA
P.O. Box 12778
No. 7 Koger Exec. Center
Suite 212
Norfolk 23502
Tel:(804) 461-4025/6

P.O.Box 9669
2914 Hungary Springs Road
Richmond 23228
Tel: (804) 285-3431

WASHINGTON
Bellefield Office Pk.
1203-114th Ave. S.E.
Bellevue 98004
Tel: (206) 454-3971
TWX: 910-443-2446

·WEST VIRGINIA
Medical/Analytical Only
Charleston
Tel: (304) 345-1640

WISCONSIN
9004 West Lincoln Ave.
West Allis 53227
Tel: (414) 541-0550

FOR U.S. AREAS NOT LISTED:
Contact the regional office
nearest you: Atlanta, Georgia...
North Hollywood, California...

Part No. 32103-90001
Printed in U.S.A. 11/74
Reprint (Update # 1 Incorporated) 9/77

HEWLETT hp PACKARD
Sales and service from 172 offices in 65 countries.
11000 Wolfe Road, Cupertino, California 95014